

CCSS OPERATING INSTRUCTIONS

C2, VOL. 15, CCSSOI 18-320
VOL. 15, CCSSOI 18-320

COMMODITY COMMAND STANDARD SYSTEM

The instructions contained in this publication become effective with implementation of Release No. 62.

ARMY AUTOMATION

**COMMODITY COMMAND STANDARD SYSTEM
OPERATING INSTRUCTIONS
(GUIDANCE)**

PROGRAMER'S HANDBOOK

MULTI-MACHINE SCHEDULER

DISTRIBUTION STATEMENT A

Approved for public release
Distribution Unlimited

THIS COPY IS A REPRINT WHICH INCLUDES
CURRENT PAGES FROM CHANGES 1 THRU 6.
CHANGED MATERIAL IS INDICATED BY A
VERTICAL BAR IN THE MARGINAL AREA.

AUTOMATED LOGISTICS MANAGEMENT SYSTEMS ACTIVITY

OCTOBER 1981

89 10 6 003

REPORT DOCUMENTATION PAGE	1. REPORT NO. DOD/SW/MT-90/001a	2.	3. Recipient's Accession No.
4. Title and Subtitle Commodity Command Standard System Operating Instructions 18-320, VOL 15 Multi-Machine Scheduler		5. Report Date Oct 81	
7. Author(s)		6.	
9. Performing Organization Name and Address Systems Integration and Management Activity P.O. Box 1578 St. Louis, MO 63188-1578		8. Performing Organization Rept. No. 10. Project/Task/Work Unit No. 11. Contract(C) or Grant(G) No. (C) (G)	
12. Sponsoring Organization Name and Address Systems Integration and Management Activity P.O. Box 1578 St. Louis, MO 63188-1578		13. Type of Report & Period Covered Supporting Documentation 14.	
15. Supplementary Notes For magnetic tape, see:			
16. Abstract (Limit: 200 words) Supporting documentation which provides technical information concerning the concepts, facilities, and use of the Multi Machine Scheduler (MMS). This instruction applies to any automatic data processing installation desiring an automated job scheduling system. Included in this documentation is a description of the complete updating and reporting capabilities available to the user for database modification with respect to job statistics, how to set up and run a simulation process, and a variety of database reports to aid those in scheduling or controlling the job processing environment. (100) 7			
17. Document Analysis a. Descriptors Automated Job Scheduling System b. Identifiers/Open-Ended Terms c. COSATI Field/Group			
18. Availability Statement No restriction on distribution. Available from National Technical Information Service, Springfield, VA 22161		19. Security Class (This Report) UNCLASSIFIED 20. Security Class (This Page) UNCLASSIFIED	21. No. of Pages 155 22. Price A08

DO NOT PRINT THESE INSTRUCTIONS AS A PAGE IN A REPORT

INSTRUCTIONS

Optional Form 272, Report Documentation Page is based on Guidelines for Format and Production of Scientific and Technical Reports, ANSI Z39.18-1974 available from American National Standards Institute, 1430 Broadway, New York, New York 10018. Each separately bound report—for example, each volume in a multivolume set—shall have its unique Report Documentation Page.

1. Report Number. Each individually bound report shall carry a unique alphanumeric designation assigned by the performing organization or provided by the sponsoring organization in accordance with American National Standard ANSI Z39.23-1974, Technical Report Number (STRN). For registration of report code, contact NTIS Report Number Clearinghouse, Springfield, VA 22161. Use uppercase letters, Arabic numerals, slashes, and hyphens only, as in the following examples: FASEB/NS-75/87 and FAA/RD-75/09.
2. Leave blank.
3. Recipient's Accession Number. Reserved for use by each report recipient.
4. Title and Subtitle. Title should indicate clearly and briefly the subject coverage of the report, subordinate subtitle to the main title. When a report is prepared in more than one volume, repeat the primary title, add volume number and include subtitle for the specific volume.
5. Report Date. Each report shall carry a date indicating at least month and year. Indicate the basis on which it was selected (e.g., date of issue, date of approval, date of preparation, date published).
6. Sponsoring Agency Code. Leave blank.
7. Author(s). Give name(s) in conventional order (e.g., John R. Doe, or J. Robert Doe). List author's affiliation if it differs from the performing organization.
8. Performing Organization Report Number. Insert if performing organization wishes to assign this number.
9. Performing Organization Name and Mailing Address. Give name, street, city, state, and ZIP code. List no more than two levels of an organizational hierarchy. Display the name of the organization exactly as it should appear in Government indexes such as Government Reports Announcements & Index (GRA & I).
10. Project/Task/Work Unit Number. Use the project, task and work unit numbers under which the report was prepared.
11. Contract/Grant Number. Insert contract or grant number under which report was prepared.
12. Sponsoring Agency Name and Mailing Address. Include ZIP code. Cite main sponsors.
13. Type of Report and Period Covered. State interim, final, etc., and, if applicable, inclusive dates.
14. Performing Organization Code. Leave blank.
15. Supplementary Notes. Enter information not included elsewhere but useful, such as: Prepared in cooperation with . . . Translation of . . . Presented at conference of . . . To be published in . . . When a report is revised, include a statement whether the new report supersedes or supplements the older report.
16. Abstract. Include a brief (200 words or less) factual summary of the most significant information contained in the report. If the report contains a significant bibliography or literature survey, mention it here.
17. Document Analysis. (a). Descriptors. Select from the Thesaurus of Engineering and Scientific Terms the proper authorized terms that identify the major concept of the research and are sufficiently specific and precise to be used as index entries for cataloging.
(b). Identifiers and Open-Ended Terms. Use identifiers for project names, code names, equipment designators, etc. Use open-ended terms written in descriptor form for those subjects for which no descriptor exists.
(c). COSATI Field/Group. Field and Group assignments are to be taken from the 1964 COSATI Subject Category List. Since the majority of documents are multidisciplinary in nature, the primary Field/Group assignment(s) will be the specific discipline, area of human endeavor, or type of physical object. The application(s) will be cross-referenced with secondary Field/Group assignments that will follow the primary posting(s).
18. Distribution Statement. Denote public releasability, for example "Release unlimited", or limitation for reasons other than security. Cite any availability to the public, with address, order number and price, if known.
19. & 20. Security Classification. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED).
21. Number of pages. Insert the total number of pages, including introductory pages, but excluding distribution list, if any.
22. Price. Enter price in paper copy (PC) and/or microfiche (MF) if known.

OPTIONAL FORM 272 BACK (4-77)

DEPARTMENT OF THE ARMY
USADARCOM AUTOMATED LOGISTICS MANAGEMENT SYSTEMS ACTIVITY
PO Box 1578, St. Louis, Missouri 63188

CCSS OPERATING INSTRUCTIONS
No. 18-320
Volume 15

8 October 1981

Army Automation

COMMODITY COMMAND STANDARD SYSTEM OPERATING INSTRUCTIONS (GUIDANCE)

PROGRAMMER'S HANDBOOK

MULTI-MACHINE SCHEDULER

	Paragraph	Page
CHAPTER 1. GENERAL		
Purpose	1-1	1-1
Applicability	1-2	1-1
Explanation of terms	1-3	1-1
General	1-4	1-2
2. SCHEDULE ALGORITHM SIMULATION SYSTEM		
Concept	2-1	2-1
SASS functions	2-2	2-1
Master data base	2-3	2-1
Execution by function	2-4	2-4
Special processing utilities and options	2-5	2-18
Coding examples	2-6	2-19
3. SCHEDULE ACTIVITY MANAGER		
Concept	3-1	3-1
SAM functions	3-2	3-1
★ Run matrix	3-3	3-2
★ Run matrix segments	3-4	3-2
Execution by function	3-5	3-5
Operational considerations	3-6	3-14
Processing considerations	3-7	3-17
4. DYNAMIC RESOURCES EXAMINATION AND MANAGEMENT SYSTEM		
Concept	4-1	4-1
DREAMS functions	4-2	4-1
Execution by function	4-3	4-2
Appendix A. SASS reports or outputs		A-1
B. SAM reports		B-1
C. List of MMS command procedures		C-1
D. List of MMS cataloged procedures		D-1
E. MMS log-on procedure		E-1
F. MMS messages		F-1
G. MMS completion codes		G-1

	Page
Appendix H. DREAMS reader information	H-1
I. ABEND codes	I-1
☆ J. MMS libraries and data sets	J-1
K. MMS modules	K-1
☆ L. MMS file information	L-1
Index	Index 1

CHAPTER 1

GENERAL

1-1. Purpose. This instruction provides technical information concerning the concepts, facilities, and use of the multi-machine scheduler (MMS).

1-2. Applicability. This instruction applies to any automatic data processing (ADP) installation desiring an automatic scheduling system.

1-3. Explanation of terms. a. MMS network—all processor and resources under the control of the master data base.

* b. Master data base—contains all information required for the scheduling or controlling of any job to be put under MMS control. The master data base consists of the job and network files.

c. Master matrix—used interchangeably with master data base.

* d. Run matrix—a data base, extracted from the master data base, that contains information required for scheduling and controlling jobs established under MMS control.

* e. On-the-fly update—the ability to modify the run matrix dynamically while jobs under MMS are executing.

f. Application—a grouping of related jobs for scheduling considerations (for example, weekly billing).

g. Job—a grouping of programs into the lowest controllable element of the system.

h. Predecessor job—one whose functional or data constraint must be met prior to another job being considered for scheduling.

i. Exclusive-use mode—a file is loaded, unloaded, or updated by one or more steps in a job under MMS control. MMS will not release other users of this file when this job is executing.

j. Share mode—a file is updated by one or more steps in a job under MMS control. MMS will release other reference users, or share users, of this file to a single central processing unit (CPU).

k. Update mode—a file is updated by one or more steps in a job under MMS control. MMS will only release other reference users of this file when this job is executing.

Paper 21.95
MF 6.95
NTIS

A-1 21

1. Reference mode--a file is read by one or more steps in a job under MMS control. MMS will release other reference, update, or share users of this file when this job is executing.

* m. Operating systems.

(1) MVT--multiple variable number of tasks (MVT).

(2) MVS--multiple tasking virtual storage system (MVS).

n. MASP--Houston Automatic Spooling Process, provides a more efficient alternative to the OS functions of input and output spooling.

o. Quiesce--to inhibit the releasing of jobs to allow the system to dry up.

p. ABEND--abnormal end (termination) of a program either with a system completion code (e.g., data check), or a user completion code, greater than 499.

q. TSO--the time sharing option used to provide remote scheduling capabilities.

*

1-4. General. a. This system allows the user to dynamically schedule externally, as well as internally, the processing of jobs within any type of MDP environment. It will perform the primary functions of effective computer use.

* b. Complete updating and reporting capabilities are available to the user including temporary and permanent data base modification, automatic data base updating with respect to job statistics, on-the-fly updating, a variety of data base reports to aid those scheduling or controlling the environment.

c. The simulation and the actual control at execution time will use the same scheduling algorithm. The only difference being that at simulation time the system resources, as defined in the data base, are used for scheduling; and at the actual time of execution, the resources within the particular computer environment are used for scheduling purposes.

* d. The system can be operated in batch mode or remotely by a terminal utilizing the TSO. The TSO option is facilitated by a special log-on procedure (\$MSTSO) and the command procedures (CLIST).

e. The system is comprised of three major components: the Schedule Algorithm Simulation System (SASS), the Schedule Activity Manager (SAM), and the Dynamic Resources Examination and Management System (DREAMS).

(1) SASS--edits input, verifies data base integrity, produces reports, simulates environments, and maintains data bases.

- (2) SAM—controls and schedules job level processing.
- (3) DREAMS—scans the job control language(JCL),
updates the run matrix resources (tapes and care),
and writes the JCL to the network job spool.

Note. These components are designed to function together or with the SASS alone, if internal control of job processing is not a requirement at a particular installation. Subsequent chapters of this document will discuss the properties of each component and their interrelationship.

CHAPTER 2

SCHEDULE ALGORITHM SIMULATION SYSTEM

2-1. Concept. a. SASS is designed to provide an effective and accurate method of automatically scheduling a computer network. It will optimize the work required based upon the environment in which the work will be performed.

b. Due to the many variables that are introduced into any scheduling task (for example, volume of input, type of input, resource availability, condition of each master file, and so forth), a minimum, or nonexistent, level of predictability relative to any given job may prevail. Hence SASS does not balance the workload based on the average running times of the jobs being scheduled. Instead it optimizes those jobs under its control based upon the maximum use of the system resources available (that is, core, disks, tapes, printers, punches, and initiators). This concept readily adapts to any dynamic environment. This even becomes more meaningful as the number of jobs to be scheduled increases and as the job relationships become more complex.

c. SASS is the front end of the MMS. For the SAM portion to be used, processing must first occur in SASS, which is the only means of accessing or updating the master data base. SASS will then, based upon the user's request, create or modify the run matrix, so that SAM can dynamically schedule the execution of each computer system.

2-2. SASS functions. The functions unique to the SASS portion of MMS are:

- a. Master data base format.
- b. Master data base update.
- c. Simulation.
- d. Schedule initialization.
- e. Master data base reports.
- f. Management reports.
- g. On-the-fly updating.

★2-3. Master data base. The master data base is the data set that SASS uses in performing its various functions. It is comprised of two files, the job file and network file. The MMS files are documented in appendix L.

★ a. Job file. This file contains all jobs that the user would want under the control of the MMS. Each job entry in this file contains all information for that job that is required by the system to perform its scheduling or controlling function. Figure A-1 depicts the elements comprising a job entry in this file. A definition of each element follows:

(1) Job number—the relative position of a job within the job file (maximum of 5760).

(2) Job name—the name of the job under MMS control.

* (3) Volume number—the application volume number of which this job is a part. This element offers a method of grouping related jobs for scheduling purposes. The volume number can be from 1 through 3400.

(4) Dependent job numbers—the data or functional constraints (maximum of 12) placed upon a job that must be met prior to that job being executed. MMS offers two different types of dependency relationships that can be indicated. One is a dependency back-off relationship, and the other is a no-back-off relationship.

(a) Back-off relationship. If, in the following job matrix segment example, only jobs A and C were to be scheduled, the resultant dependency relationship would appear as shown in the resultant job matrix. Since job C's dependency was job B, and job B was not scheduled, then job C would back-off its dependency toward a job that was in this same string and was scheduled, which in this example was job A.

Example:

<u>Job matrix segment</u>		<u>Resultant job matrix</u>	
<u>Job</u>	<u>Dependency</u>	<u>Job</u>	<u>Dependency</u>
A	None	A	None
B	A	C	A
C	B		

(b) No back-off relationship. If, in the following job matrix segment example, only jobs A and C were to be scheduled, the resultant dependency relationship would appear as shown in the resultant job matrix. Note the asterisks surrounding job C's dependency. The asterisk indicates the no-back-off relationship. In this example, since job B was not scheduled and job C's dependency of job B indicates no-back-off, the dependency of job C on the resultant matrix becomes NONE.

Example:

<u>Job matrix segment</u>		<u>Resultant job matrix</u>	
<u>Job</u>	<u>Dependency</u>	<u>Job</u>	<u>Dependency</u>
A	None	A	None
B	A	C	None
C	*B*		

(5) Region—the region that the job will request.

* (6) Files used—the files (maximum of 12) that the job uses and the modes (R=reference, U=update, S=share, E=exclusive) in which they are used.

- (7) Tape 9—the number of 9-track tape devices the job uses.
- (8) Tape 7—the number of 7-track tape devices the job uses.
- (9) Tape D—the number of dual density tape devices the job uses.
- (10) Tape 6—the number of 6250 BPI tape devices the job uses.
- (11) Print—the number of printers the job directly allocates.
- (12) Punch—the number of punches the job directly allocates.
- (13) Identify—classified or privacy job.
- (14) Recent run date—the Julian date of the most recent running of this job (automatically posted).
- (15) Recent run time—the time in minutes that the most recent running of this job required (automatically posted).
- (16) Average run time—the time in minutes of the current average run time of this job (automatically updated).
- (17) Times run—the number of times that this job has been run since the installation of MMS (automatically updated).
- (18) Priority—indicates whether or not this job is deemed relatively more critical than other jobs in the job matrix segment. It designates the job's relative priority for scheduling purposes; it must be two numeric digits from 15 through 01 (default for ADD is 02).
- (19) Simulation flag—indicates whether or not a job is to be scheduled for simulation (used for internal processing only).
- (20) Identify machine(s)—indicates whether or not a job can process on a machine.
- * (21) Early start time—the time when this job is to be first considered for scheduling (from 0001 through 2400).

b. Network file. The network file is comprised of six distinct segments. A definition of each segment follows:

- * (1) Application segment. This segment contains the application mnemonics for each of the application volume numbers that the user has established. This segment offers the capability of grouping related jobs for scheduling purposes. The application volumes are cross-referenced to the application mnemonics and vice versa. Figures A-2 and A-3 reflect this double cross-reference. The application volume numbers must be from 1 to 999.
- * (2) Device segment. This segment contains tables of devices (that is, disks, tapes, printers, and punches) that the user defines to MMS for simulation purposes only. These tables are not used by SAM; SAM actually scans the computer system to make a point-in-time determination of what resources are available and has no need to interrogate these tables. Figure A-6 depicts the device table and its elements.

★ (3) File segment. This segment contains the file mnemonics (and file types) that the user has established. There is a double cross-reference from file number to file mnemonic and vice versa. Figures A-4 and A-5 reflect this double cross-reference. The file numbers must be from 1 to 255.

★

(4) Installation segment. This segment contains the 8-byte installation mnemonic assigned by the user and is displayed on all SASS reports.

★ (5) Machine network segment. This segment contains the machine identification (ID), time available, and tape usage for each CPU defined to MMS.

★ (6) File control segment. This segment contains the time, date, and CPU ID of last update to the master data base.

2-4. Execution by function. a. General The execution of the various functions under SASS is done by cataloged procedures (PROCs) as shown in appendix B and command procedures (CLISTs) as shown in appendix C. Each function has one or more associated PROCs and may have an associated CLIST. The symbolic parameters, as depicted in appendixes B and C, are assigned the default values to be used in the production environment, therefore, need not be repeated in the execution PROCs or CLISTs. The only symbolic parameters required at execution time are those mentioned in the narrative on function definition.

b. Input data set. When required, the input control data set for SASS PROC and CLISTs will contain one or two types of statements:

(1) Control card (statement). This statement defines the operation being performed and the master data base segment addressed (or the type report requested). The operation and operand must be on the same card (statement). The format is as follows:

<u>cc 1-3</u>	<u>Operation</u>	<u>Operand</u>
./b	operation	operand

Note. Past column 3, free form coding prevails.

(2) Data card (statement). This statement(s) is optionally present based upon the operation field of the control card. It defines the input to SASS to perform data base creation, maintenance, or simulation. All data on this statement are free form, keyword coding with the exception of data continuing (relative to one action of the operation) on no more than one card. The continuation rules are as follows:

(a) A comma must be inserted after the last keyword data combination on a card.

(b) One or more blanks are required prior to column 72.

(c) Continuation of data on subsequent cards must begin after column 1.

c. Function definition. The definition and performing characteristics of each function and the execution JCL required are as follows:

★ (1) Master data base format. This function creates and formats the master data base for subsequent update processing and is used for the initial installation of the MMS only.

(a) PROC execution.

```
//stepname EXEC SASFMTP,MMATRIX=XXXXXXXX,NEWVOL=YYYYYY
```

(b) CLIST execution.

★ SASFMT

(2) Master data base update. This function updates segments within the master data base either temporarily or permanently. The execution of this function will generate a report of each of the segments updated indicating whether the change was temporary or permanent.

(a) Master data base update.

Execution JCL.

```
//stepname EXEC SASSUPDP
```

(b) Input control data set. The input data set is as follows:

1 Control card.

<u>cc 1-3</u>	<u>Operation</u>	<u>Operand</u>
./b	PMOD	MATRIX
	TMOD	APPL
		FILES
		DEVICES
		ENV
		NETWORK

★

Note. Permanent modifications (PMODs) and temporary modifications (TMODs) are mutually exclusive operations. Only one type of operation may be performed for each SASS execution. However, each execution may address more than one of the segments (operands) of the data base using the same type operation.

2 Data cards. The data cards must immediately follow the PMOD or TMOD control card for the segment of the master data base being addressed by the operand. Free form keyword coding may begin in any column prior to 72. Continuation rules are followed (para 2-4b(2)). Keywords must be separated by a comma. Keywords are separated into three categories: required for all operands, required for specific master data base segments, and optional for specific segments.

a Required keyword for all operands. This keyword is required to define the type of update to be performed.

<u>Keyword</u>	<u>Field content</u>	<u>Description</u>
ACTION= AC	ADD	A new entry, one that does not presently exist, is to be added to the specific segment identified by the control card operand; it cannot add fields to an existing entry.
	CHANGE	Field(s) are to be altered in an existing entry of the specific segment identified by the control card operand.
	DELETE	An existing entry is to be deleted from the specific segment identified by the control card operand.

b Required keywords for specific segments. These keywords are required to obtain entry into the designated segment.

<u>Segment</u>	<u>Keyword</u>	<u>Description</u>
★ Job matrix	JOENUM = JN	The relative number of the job within this segment, it must be one to four numeric digits from 1 through 5670.
★ Application	APPLNO = AN	The application volume number; it must be four numeric digits from 1 through 3400.
Files	FILENO = FN	The file number of the file being addressed; it must be one to three numeric digits from 1 through 255.
Devices	None	All keywords are optional.
Environment (includes dummy UCBs, core, and installation segments)	None	All keywords are optional.

c Optional keywords for specific segments. These keywords are to be used at the discretion of the user to update those fields requiring modification.

<u>Segment</u>	<u>Keyword</u>	<u>Description</u>
Job matrix	JOBNAME = JM	The job name of the job being modified; can be up to eight characters; <u>required</u> if the ACTION keyword specifies ADD since there is no default assignment; required if the ACTION keyword specifies DELETE.
	APPLNO = AN	The application volume number; must be four numeric digits from 1 through 3400; <u>required</u> if the ACTION keyword specifies ADD, since there is no default assignment.
	DEPNO = DP	The job number(s) that places a data and/or functional constraint on the job being modified, each must be one to four numeric digits from 1 through 5760; must be enclosed in parentheses even if only one job is a constraint or NONE is specified; a maximum of 12 dependent job numbers may be specified; if the no back-off option is used, the job number must be preceded by a minus (-) sign; the dependency numbers are treated as an AND condition, not as an OR condition, therefore, all dependent jobs must complete prior to this job being scheduled; default assignment for ADD is NONE.
	CORE = CR	The region required to execute this job, must be one to four numeric digits; default for ADD is 54K.
	PRIORITY = PR	The indicator designating the job's relative priority for scheduling purposes; runs from the high of 15 to a one; and must be two numeric digits from 15 through 01, default for ADD is 02.
☆	FILENO = FN	The file number(s) designating the files used by this job; each must be one to three numeric digits from 1 through 255 preceded by an R (reference), U (update), E (exclusive), or S (share); must be enclosed in parentheses even if only one file is used or NONE is specified; a maximum of 12 file numbers may be specified; default for ADD is NONE. See note for more information.

<u>Segment</u>	<u>Keyword</u>	<u>Description</u>
Job Matrix	TAPE9 - T9	The number of each type of device required for this job; must be one or two numeric digits; default for ADD = 0.
	TAPE7 - T7	
	TAPED - TD	
	TAPE6250 - T6	
	PRINT - PT	
	PUNCH - PC	
	AVETIME - AT	The average running time in minutes for this job; must be one to four numeric digits; default for ADD=0; automatically maintained when times run is greater than 9, times run is automatically reset to one.
	TIMESRUN - TR	
	RECDATE - RD	
	RECTIME - RT	
	CLASPCY - CL	
★	EARLYST - ES	The early start time for a job (from 0001 through 2400) will place an early start time constraint for that job. This time allows a 23 hour 59 minute look-ahead and indicates when the job is to be first considered for scheduling.

<u>Segment</u>	<u>Keyword</u>	<u>Description</u>
	MACHDEP = MD	The machine(s) the job must run on must be eight alphabetic positions. If the job must run on machine one or three, enter (YNYNNNNN); required if the ACTION keyword specified ADD, since there is no default assignment.
★ Application	APPLNAME = AM	The application mnemonic for the specified application number; must be one to seven characters; no default is assigned; always required.
Files	FILENAME = FM	The file mnemonic for the specified file number; must be one to five characters; <u>required</u> if ACTION is ADD, since there is no default assignment; required if ACTION is DELETE.
	TYPE = TY	The type of file: V--VSAM M--CCSS DMR file S--S2K O--other Default is M

★Note. File usage by MMS is defined as follows:

Reference--file is only being read.

Update--file is being updated and other users can reference it.

Exclusive--file is being updated but no other use is allowed.

Share--used for DMR files only. File is being updated and the DMR access routines control updating by MAJOR key. File being shared can have other share and/or reference users.

Devices	TAPE9 = T9	The number of each type of device to be identified to SASS for simulation purposes; must be one or two numeric digits; no default assignment is made at data base creation time; ACTION keyword must specify CHANGE.
---------	---------------	--

TAPE7 =
T7

TAPED =
TD

TAPE6250 =
T6

<u>Segment</u>	<u>Keyword</u>	<u>Description</u>
	DISK = DK	
	PRINT = PT	
	PUNCH = PC	
Environment	CORE = CR	The total available system core in thousands (used for simulation only); must be one to four numeric digits from 2 to 8000; ACTION keyword must specify CHANGE.
	INSTL =	The installation mnemonic; must be one to eight characters; ACTION keyword must specify CHANGE.
★ Network	NTWRKNO = NN	The network number for each CPU as it is defined to the scheduler; must be one numeric digit with a value of 1 through 8; required keyword; ACTION keyword must specify CHANGE or DELETE.
	MACHID = MI	The machine ID of the network machine being defined; must be the first two digits of the systems management control area (SMCA); ACTION keyword must specify CHANGE.
	AVSTIME = AS	The beginning start time that this machine will be available to the scheduler for batch processing; must be four numeric digits representing HHMM (from 0001 through 2359); the default is 0000 meaning that this machine will always be available for batch processing; ACTION keyword must specify CHANGE.
	AVETIME = AE	The time that this machine will no longer be available for batch processing, must be four numeric digits with same criteria as available start time above; must be used in conjunction with AVSTIME; ACTION keyword must specify CHANGE.

<u>Segment</u>	<u>Keyword</u>	<u>Description</u>
★	TAPEDUSE - DU	Define to the scheduler whether or not TAPED can also be used as a TAPED for this machine; must be either Y or N; default is N; ACTION keyword must specify CHANGE.
★	TAPE6USE - 6U	Define to the scheduler whether or not TAPE6 can also be used as TAPE9 for this machine; must be either Y or N; default is N; ACTION keyword must specify CHANGE.

★ (3) Simulation. The simulator portion of SASS can be used to simulate the processing of applications and/or jobs as if those jobs were to be executed on a computer utilizing a multiprogramming environment. The simulator considers factors such as available core and devices (disk, tape, printers, punches), active and inactive initiators, priority, disk file usage, and so forth. Jobs and/or applications can be scheduled, any temporary or permanent modification to the master data base can be made, and special keyword processing can be utilized for performing specific actions. Messages are produced indicating reasons why a job could not be released to be used for rescheduling and reconfiguring purposes. This function will not produce a run matrix; therefore simulation and re-simulation of varied environments can be accomplished without impact.

(a) The execution of this function will generate three reports:

1 Jobs being scheduled. This report will depict the jobs being scheduled (simulated) and the characteristics of each job, including the dependencies, after processing by the SASS dependency optimizer. Figure A-13 shows the elements within this output product.

2 Inter-intra application dependencies. This report breaks out the inter and intra application dependencies by job within an application for just those jobs being simulated. This is primarily for use by scheduling personnel, when jobs must be scheduled out of their normal scheduling sequence. Figure A-14 shows the elements within this output product.

★ 3 Hardcopy simulation. This report depicts the running of the scheduled jobs in a multiprogramming environment. It will indicate idle time, if present. Also, at the end of the report, a summary of the times are shown (elapsed time, sequential time, idle time). Figure A-15 shows this type of output product.

(b) Execute the simulation function.

PROC execution.
//stepname EXEC SASSIMP

(c) Input control data set. The input data set is as follows:

1 Control card.

<u>cc 1-3</u>	<u>Operation</u>	<u>Operand</u>
★ . /b	RUN	b/HMM

Note. A four-position time representing HMM may be used in the operand to represent a beginning time for the simulation report.

★ 2 Data cards. Cards must immediately follow the RUN control card. Free form coding may begin in any column prior to 72. Continuation rules (para 2-4b(2)) are followed. Keywords must be separated by a comma. The first 555 jobs defined to the simulator will be scheduled; any others will be eliminated from the scheduling period, a corresponding message will be generated, and the simulation will be terminated. Duplicate keywords for APPLNAME and JOENAME are allowed and treated as an add-on for the scheduling period. An asterisk (*) preceding a specific jobname or application name being defined to the simulator will assign priority to that job or all the jobs within the specified application. A slash (/) followed by a four-digit time (from 0001 through 2400), following a jobname or application name places an early start time constraint on that job or all jobs within the specified application. This entry will override the early start time on any affected job. This time allows a 23 hour 59 minute look-ahead and indicates when the particular job(s) is to be first considered for scheduling. All keywords are optional with the exception that at least one job must be input to the simulator for scheduling by either the APPLNAME or JOENAME keyword.

<u>Keyword</u>	<u>Description</u>
APPLNAME= AM	Each application mnemonic being defined to the simulator; must match the application mnemonic defined in the application segment of the master data base (para 2-3b); must be enclosed in parentheses, even if only one application is input.

<u>Keyword</u>	<u>Description</u>
JOENAME = JM	Each job name being defined to the simulator that is not in a unique functional application, or is the only job of many within an application that is to be scheduled, or is the only job within an application already defined by APPLNAME to which the user will assign a priority or an early start time; must match the job name as defined in the job matrix segment of the master data base (para 2-3a); must be enclosed in parentheses, even if only one job is input.
INIT = IN	The number of initiators defined to the simulator for use during this simulation; used only within the simulator; must be one numeric digit from 1 through 10; default assignment is eight initiators.
STRTIME = ST	The time of day representing the starting time for the simulator during this simulation; must be one to four digits from 000 through 2400; default assignment is the current time of day.
INTERVAL = IT	The time increment in minutes to be used during the simulation; must be one or two digits from 1 through 60; default assignment is three minutes.
MESSAGE = MS	A message that will appear on a SAM report in the upper right corner; replaced each time a SASSRUN, SASSSIM, SASSCHK, or SASSFLY function is executed; can be used for cycle identification; default assignment is blank.

★ (4) Schedule initialization. SASS is used to initially start up a scheduling period by creating the run matrix that supports those jobs scheduled for execution. This function performs the same tasks as the simulator (para 2-4c(3)) with the additional tasks of creating the run matrix and starting the release of jobs to the system. Jobs that are stored on the EXECUTION JCL file are automatically copied to the JCL job spool.

(a) Initiate the schedule.

PROC execution
//stepname EXEC SASSRUNP

★ (b) Input control data set. The input data set is as follows:

Control card

<u>cc 1-3</u>	<u>Operation</u>	<u>Operand</u>
./b	RUN	b

(5) Master data base reports. SASS produces three types of reports as follows:

(a) SASS standard reports. This option will print selected (or all of the) master data base segments and cross-references between segments. The reports are depicted in figures A-1 through A-10.

1 PROC execution.
//stepname SASSRPTP

2 Input control data set. The input data set is as follows:

a Control card.

<u>cc 1-3</u>	<u>Operation</u>	<u>Operand</u>
./b	REPORT	MATRIX APPL FILES RESOURCES NETWORK ALL XMATRIX XAPPLJOB XAPPLDEP FILEMODE

b Data card(s). No data cards are permitted as input to this PROC.

c Control card conventions. Only one operand may be specified per report control card. Multiple report control cards may be input.

d Report operand definitions:

MATRIX--generates a report of the job matrix segment in job number sequence (fig A-1).

APPL--generates a double cross-reference of application volume to application mnemonic (sample report in fig A-2), and application mnemonic to application volume (fig A-3); not produced under TSO.

FILES--generates a double cross-reference of file number to file mnemonic and file mnemonic to file number (figs A-4 and A-5).

RESOURCES--generates a report showing the system resources available for simulation (core and devices) (fig A-6).

XMATRIX--generates a report of the job matrix segment in job name sequence (fig A-7).

XAPPLJOB--generates, in application mnemonics sequence, a cross-reference of all jobs within each application (fig A-8).

XAPPLDEP--generates, in job within application sequence, a breakout of the inter and intra application job dependencies (fig A-9).

FILEMODE—generates, in file mnemonic sequence, a report of all jobs using that file and a breakout of the mode used (reference, update, share or exclusive) (fig A-10 (app A)).

ALL—generates all reports produced by the above operands.

(b) **SASS cross-reference reports.** This program will print master data base dependency cross-reference data pertaining to an application name, number or job name (fig A-16 through A-18 (app A)). It will also produce in file acronym sequence cross-reference reports between files and update exclusive-user, and users (fig. A-19, app A).

PROC execution
//stepname EXEC SCXREFP,FUNKSHN=XX

Where 'XX' specifies the cross-reference to be produced.

FUNKSHN=JM report in job name sequence of job dependencies and dependent jobs.

FUNKSHN=AM report in application name sequence of job dependencies and dependent jobs.

FUNKSHN=AN report in application number sequence of job dependencies and dependent jobs.

FUNKSHN=FA report in file acronym sequence between files and update, exclusive use, and share users.

(c) **SASS inquiry.** This program will display master data base information by application name(s), application number(s), job name(s), or job number(s).

CLIST execution.

Display SASS information for specified application name(s).
SCAMINQ

response *** ENTER APPLICATION NAME(S) (MAX of 50) ***
response *** WHEN THROUGH ENTER /* ***

Display SASS information for specified job name(s).
SCJMINQ

response *** ENTER JOB NAME(S) (MAX of 50) ***
response *** WHEN THROUGH ENTER /* ***

Display SASS information for specified job number(s).
SCJNINQ

response *** ENTER JOB NUMBER(S) (MAX of 50) ***
response *** WHEN THROUGH ENTER /* ***

Display SASS information for specified application volume number(s)
SCANINQ

response *** ENTER APPLICATION VOLUME NUMBER(S) (MAX of 50) ***
response *** WHEN THROUGH ENTER /* ***

(6) On-the-fly update test. The running matrix is updated during production (on-the-fly) by first executing a test of the changes to be performed.

(a) Execution considerations. All accepted input to the SASSSIMP and SASSUPDP PROCs can be input (i.e., updates, job adds/deletes to the master data base) with the additional capability of job/application adds/deletes to the running matrix. The execution of the PROC for this function, SASSCHKP, will generate a report of the before and after image of the running matrix (fig. B-1) though no actual update will have taken place.

(b) Restrictions. Only TMODs are allowed as input. Any prior dependency TMOD to a job on the current running matrix must have a TMOD applied again. TMODs affecting any fields other than dependencies need not be repeated as SASS will recognize them. Any job that was a TMOD ADD to the current running matrix need not be repeated as SASS will recognize it as such and accept all fields, including the dependencies.

(c) Test an on-the-fly update.

PROC execution
//stepname EXEC SASSCHKP

(d) Sequence of processing.

1 Any job that has started and ended normally will be posted to the master data base. Any job that has previously ABENDED will not be posted.

2 Any job that has started and ended normally, including any job that has previously ABENDED but has now completed normally, will be deleted from the running matrix.

3 All remaining jobs on the running matrix are flagged as input for the simulator and scheduling algorithm.

4 The input data set (CARDIN) is read.

5 All TMODs are processed.

6 All FLY DELETES are processed.

7 All FLY ADDs are processed.

8 The simulator is activated.

9 No running matrix or master data base rewrites (updates) are accomplished.

(e) Input control data set. The input data set is as follows:

1 Control card.

<u>cc 1-3</u>	<u>Operation</u>	<u>Operand</u>
./b	FLY	ADD
		DELETE

2 Data card(s). Conventions previously mentioned for the SASSSIMP PROC (para 2-4c(3)(c)2) apply for this function. Applications and/or jobs may be deleted or added to the running matrix.

* (7) On-the fly update. Processing, control cards, data card(s), and outputs are the same as those for the on-the-fly update test (SASSCHKP) (para 2-4c(6)), with the exception that this function physically updates the running matrix and the master data base. Jobs that are added to the running matrix and are on the execution JCL file will be written to the network job spool.

(a) Perform an on-the-fly update.

PROC execution
//stepname EXEC SASSFLYP

(b) Input control data set (para 2-4c(6)(e)).

d. Function usage cross-reference. Many of the SASS data base functions (referenced by their PROC names) can be used together; however, there are exceptions. Some functions are mutually exclusive. Table 2-1 illustrates what function's input control data set may be secondary input to other functions.

Add. 2-1. Allowable fu

ps

Function NumberFunction PROC name

1	SASSFMP
2	SASSUPDP
3	SASSSIMP
4	SASSRUNP
5	SASSEPTP
6	SASSCHKP
7	SASSFLYP

SECONDARY

<u>PRIMARY</u>	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>
1		N	N	N	N	N	N
2	N		N	N	Y	N	N
* 3	N	*		N	Y	N	N
* 4	N	*	N		Y	N	N
5	N	N	N	N		N	N
6	N	*	N	N	N		N
7	N	*	N	N	N	N	

Y -- allowable.

N -- not allowable

* -- only TMOD allowable.

2-5. Special processing utilities and options. a. Compare utility. This utility will allow the user to compare any two master data bases. The output will indicate only those changes encountered by the utility.

PROC execution

//stepname EXEC SCCOMPRP,SASS1=XXXXXXXX,SASS2=YYYYYYYY

where XXXXXXXX = DSN of any master data base.

where YYYYYYYY = DSN of any master data base.

b. Infinite dependency set-up. This option is available when the situation occurs in which more than 12 dependencies are required for one job. Note that the dummy jobs are not scheduled, but are a permanent entry in the master data base.

(1) Technique.

(a) Establish dummy jobs in the job matrix segment; enough to contain all those dependencies required (should indicate the no-back-off option) for the job in question.

(b) Create dependencies for the job in question pointing to the dummy jobs (should indicate the back-off option).

* (2) Example. In this example, JOBA will effectively have 26-dependent jobs. If any of those 26 dependent jobs are scheduled in this scheduling time frame, JOBA will run only after they complete.

Job matrix segment

<u>Job number</u>	<u>Job name</u>	<u>Application number</u>	<u>Dependencies</u>
100	JOBA	427	801,802,803
801	DUMBY1	600	-10,-11,-12,-13, -14,-15,-16
802	DUMBY2	600	-301,-302,-303, -304,-305,-401, -402,-403,-404,
803	DUMBY3	600	-511,-512,-513, -514,-515,-516, -517,-518,-519, -520

c. Distribution instructions.

(1) Several of the SASSMAT PROCs contain an IEBGENER step which is used for the input of distribution instructions. They are SASSCHKP, SCXKFP, SASSFLYP, SASSRPTP, and SASSRUNP.

(2) To use this option, the user must override the input data definition (DD) statement to specify a data set containing the user's distribution instructions. The PROCs default to DD DUMMY,DCB=BLKSIZE=80. The user's data set must contain fixed length 80 byte records. The stepname to override is DIST. The DDNAME to override is SYSUT1.

2-6. Coding examples. a. Formatting a master data base (for initial build).

```
//stepname EXEC SASSFMT,MMATRIX=MMATRIX,NEWVOL=VOL001
```

b. Updating segments in the master data base.

```
//stepname EXEC SASSUPDP
./ PMOD MATRIX
  AC=ADD,JN=825,JM=NEWJOB,AN=600,
    DP=(NONE),CR=76,T9=4,MD=(YNNNNNNN)
  AC=CHANGE,JN=10,DP=(-723,265,11),T9=4
  AC=DELETE,JN=101,JM=X293
./ PMOD APPL
  AC=ADD,AN=600,AM=NEWAPPL,MD=(YNNNNNNN)
  AC=CHANGE,AN=524,AM=CHGAPPL
  AC=DELETE,AN=427,AM=DELAPPL
./ PMOD FILES
  AC=ADD,FN=60,FM=FILEX,PK=2,MD=(YNNNNNNN)
  AC=CHANGE,FN=3,PK=14
  AC=DELETE,FN=4,FM=MFDEL
./ PMOD ENV
  AC=CHANGE,CR=1000
```

```
* ./ PMOD NETWORK
  AC=CHANGE,NN=1,MI=F1,DU=Y,64=Y
```

c. Simulating an environment.

```
//stepname EXEC SASSSIMP
./ RUN
  AM=(BILLING,*PAYROLL,ACCTREC),
  JM=(*NEWJOB/1400),
  IN=4,IT=5,
  MS=TESTRUN
./ TMOD MATRIX
  AC=ADD,JN=845,JM=NEWJOB,AN=600,
    DP=(-4,6,725),CR=60,T9=1,MD=(YNNNNNNN)
  AC=CHANGE,JN=6,CR=142
```

d. On-the-fly updating.

```
//stepname EXEC SASSFLYP
./ FLY ADD
  A=(*APPL1,APPL2/1600),J=(JOB1),
  MS=FLYTEST1
./ TMOO MATRIX
  AC=CHANGE,JN=623,DP=(14,-15,163)
  AC=CHANGE,JN=625,FN=(U4,R5,S6)
  AC=CHANGE,JN=342,PR=1
./ FLY DELETE
  A=(APPL3),J=(JOB2)
```

e. Initiating a schedule.

```
//stepname EXEC SASSRUNP
./ RUN
  A=(HEARING,INTERACT,CATCANT,
    VESILL),
    J=(*A402)
./ TMOO MATRIX
  AC=CHANGE,=C402,FN=(RC3,S04,R10,UC5,
    .UC6,UC1,UC2)
```

f. Producing standard reports.

```
//stepname EXEC SASSRPTP
./ REPORT XMATRIX
./ REPORT FILEMODE
```

g. Producing cross-reference reports.

* //stepname EXEC,FUNKSHH=AM

CHAPTER 3

*

SCHEDULE ACTIVITY MANAGER

*3-1. Concept. a. Scheduling is designed to automatically control the execution of scheduled jobs in the MMS network for a specified period. This control is achieved by means of one running matrix that defines all jobs to be run, the characteristics of each job (e.g., files used, devices required, etc.), and the dependencies or relationships of those jobs with each other. All of these factors along with available system resources (e.g., initiators and devices) as they exist at that point in time are evaluated by MMS to effect an optimum network environment.

b. The available system resources are determined by MMS through a scan of a specific system at each point in time that scheduling occurs. The system resources that are defined in the master data base are not used by SAM as the scan of the computer system provides an exact picture at that instant.

c. Everything to be processed on a computer system need not be within the master data base (and consequently in the running matrix). Since SAM recognizes the total resources in use in each computer system, it is aware of any resources being used by a job not under its control. SAM will continue to optimize any jobs under its control and schedule these jobs accordingly based on total resource use within each system.

* d. When SAM releases a job, it is certain that the job will begin execution, due to the system resources scan. This is relevant in eliminating stacking of jobs in the job queue. Job stacking would de-optimize anything that SAM would have optimized. This no-stacking concept will effect better throughput and hence, more effective use of each computer system.

e. SAM will check three areas for a job if all constraints and resources have been met. They are checked in the following order: hold queue, network job spool, and then permanent JCL. If the job is not located in one of the three areas, no further check is made.

3-2. SAM functions. There are several functions unique to SAM processing available to the user to monitor and/or initiate action within a system. These functions include:

- a. Run matrix status.
- b. System environment scan.
- c. Job restart.

★ d. Inhibit installation scheduling.

★ e. Restart installation scheduling.

f. Individual system termination/restart.

★

g. Simulation of current workload.

h. Posting job completion.

i. System scan.

j. Running matrix update.

k. Display jobs on job network spool.

l. Cancel job(s) on job network spool.

★3-3. Run matrix. a. The run matrix is the data set that SAM uses for scheduling or controlling the network environment.

★ b. The run matrix can be created or modified by the SASS portion of MMS. This forced interaction with SASS will ensure the integrity of the master data base and the run matrix and, hence, the total system, as SASS has the ability to intercept possible error conditions prior to their application to the run matrix.

★ c. The run matrix can also be modified by the SAM portion of MMS. This method offers more flexibility in modifying the run matrix; however, there is no interaction with the SASS. The user accepts total responsibility for any run matrix additions, changes, or deletions, when using this option.

★3-4. Run matrix segments. Figure B-1 depicts the segments in the run matrix that are used by SAM to schedule or control the system. A definition of each segment and its elements follows.

★ a. Job segment. This segment contains all jobs that are currently scheduled for execution during a processing cycle.

★ (1) Run job number—the run matrix job number; the relative position within the run matrix of a job under MMS control (maximum of 630).

(2) Master matrix job number--the master data base job number; corresponds to the relative position of the job within the master data base.

(3) Job name--the name of the job under MMS control.

★ (4) Priority--indicates a job's relative scheduling importance within the run matrix (value of (15) to (81)).

(5) Job start--flag indicating whether the job has started (that is, posted with an S).

(6) Job end--flag indicating whether or not a job has ended (posted with an E), has abnormally ended (posted with an A), or has been flushed (posted with an F).

(7) Start date--the four-position Julian date that the job started.

(8) Start time or early start time--the time of day that the job started or the time of day that the job may be considered for scheduling.

(9) End time--the time of day that the job ended or abnormally ended.

(10) Early start time--the time the job will first be considered for releasing.

(11) Elapsed time--the total processing time of a job, includes accumulated time of a job if it abnormally ended, restarted, and ended (or abnormally ended again).

(12) Average time--the average elapsed run time of the job.

(13) Job ABEND counter--the number of times that a job has abnormally ended.

(14) Application name--the application name of the job.

(15) Application number--the application volume number of the job.

(16) Classification--the security or privacy classification of the job.

★ (17) Dependent job numbers--the data and/or functional constraints (maximum of 12) of a job, extrapolated from the master data base, in terms of the run matrix job numbers.

(18) Region--the region required to execute the job.

(19) Tape 9--the number of 9-track tape drives required for the job.

(20) Tape 7--the number of 7-track tape drives required for the job.

(21) Tape D--the number of dual density tape drives required for the job.

(22) Tape 6--the number of 6250 BPI tape drives required for the job.

(23) Print--the number of printers required for the job.

(24) Punch--the number of punch units required for the job.

★ (25) Files used--the files (maximum of 12) that the job uses and modes (R=reference, U=update, S=share, E=exclusive) that are used.

(26) Machine dependencies--the machines job can execute on.

(27) Machine run--machine number job is currently executing on or completed on.

★ b. File segment. This segment contains the status, number of current users, and if the file status is shared, and the CPU that the file is being used on. There are 255 file entries. The relative position of the master file entry in this segment corresponds to the file entry in the data base file segment (app B).

c. File status. The file status is the current status of the master file. The values and corresponding definitions are:

<u>Value</u>	<u>Definition</u>
Low-value	File is not in use.
U	File is updated by one job.
S	File is shared by one or more jobs.
E	File is exclusively used by one job.
L	File is locked or unavailable for use.

★ d. Termination segment. This segment consists of a one-byte field designating whether or not the run matrix is termed (quieced). Y denotes that the run matrix is termed; N denotes that the run matrix is active.

★ e. Message segment. An eight-byte field that is displayed on all run matrix reports. This value is set during the execution of the SASSRUN or SASSFLY function.

f. Date and time segment. The date and time that a SASSRUN or SASSFLY function was last executed.

g. Installation segment. This segment contains the eight-byte installation mnemonic. This value is set during the execution of the SASSRUN or SASSFLY function.

★ h. Network segment. This segment contains information on how the machines are defined to the scheduling network. It consists of machine ID, the starting and ending time each machine is available to the scheduler, and defines whether or not TAPED and TAPE6 can be used as a TAPE9. Up to eight network machines can be defined. The relative position corresponds to the eight-byte machine dependency entry in the file segment (app B).

3-5. Execution by function. a. General. The execution of the various functions under SAM is done by PROCs as shown in appendix D, and CLISTs as shown in appendix C. Each function has one or more associated PROCs and CLISTs. The symbolic parameters, as depicted in appendixes C and D are assigned the default values to be used in the production environment; therefore, need not be repeated in the execution PROCs or CLISTs. The only symbolic parameters required at execution time are those mentioned in the following narrative.

b. Execution options.

(1) Selected PROCs associated with SAM processing can be initiated into the system in any of three methods.

(a) Through an EXEC card (for example, // EXEC procname).

(b) Through the system reader (for example, // S procname).

(c) Through the system console (for example, S procname).

(2) Each CLIST associated with SAM processing is initiated through the terminal (for example, clistname) using TSO.

c. Function definition. Each of the functions under SAM, the definition and performing characteristics of each function, and the JCL required to execute each function are:

★ (1) Run matrix status. This function obtains current status of the run matrix; it can be run at any point in time and does not modify either data base. (Fig B-1, illustrates this output report.)

★ (a) Display status of total run matrix.

```
PROC EXECUTION
//STEPNAME EXEC SAMRPTP
```

```
CLIST EXECUTION
SAMRPTA
SAMRPTAH
```

- ★ (b) Display status of total run matrix in job name sequence.

```
PROC EXECUTION
//STEPNAME EXEC SAMRPTP,FUNKSHN=RPTJBA
```

```
CLIST EXECUTION
SAMRPTJ
SAMRPTJH
```

- (c) Display status of only those jobs that have not been successfully completed.

```
PROC EXECUTION
//STEPNAME EXEC SAMRPTP,FUNKSHN=RPTINC
```

```
CLIST EXECUTION
SAMRPTI
SAMRPTIH
```

- (d) Display status of only those jobs that have not been successfully completed in job name sequence.

```
PROC EXECUTION
//STEPNAME EXEC SAMRPTP,FUNKSHN=RPTJBI
```

```
CLIST EXECUTION
SAMRPTK
SAMRPTKH
```

- (e) Display status of only those jobs that have start and ABEND flags.

```
CLIST EXECUTION
SAMRPTR
```

To select a SAM report by system(s), add parameter SYS=XXX or SYS(XXX) to the execution. XXX represents from one to three processors to be displayed. Default is all.

Note. Output is directed to hardcopy.

(2) System environment scan. This function obtains the current status of the system environment as it pertains to core, disks, tapes, printers, punches, and initiators; it can be run at any point in time and does not modify either data base. (Fig R-2 illustrates this output report.)

```
PROC EXECUTION
//STEPNAME EXEC SASENVP
```

```
CLIST EXECUTION
SASENV
```

(3) Job restart. This function is used to restart a job that previously has abnormally terminated. It clears out the job start and end flags and the start date, start time, and stop time fields; eliminates job flush postings (F) for jobs dependent on the abnormally ended job; removes file lock(s) if the abnormally ended job used file(s) in update, exclusive use, or share mode; and initiates a scan of the running matrix to determine optimum job(s) to release to the system.

```
PROC EXECUTION
//STEPNAME EXEC SARESTP,JN=XXXX
```

```
CLIST EXECUTION
SAREST JN(XXXX)
```

Where XXXX = the master matrix job number (from one to four numeric digits) of the job to be restarted.

* (4) Job start. This function is used to start only the job specified. It will also start a job that has abnormally ended and will function similar to a restart except that no scan is initiated to release more jobs. A job start can be accomplished even if the system is quiesced.

(5) System termination. This function is used to quiesce the system. It inhibits the releasing of jobs under MMS control, must be used if MMS is active, and the system can be reinitiated only by SAMRUN function. To quiesce a selected processor, use SAMUPD data entry RM.

```
PROC EXECUTION
//STEPNAME EXEC SAMTERM
//          S      SAMTERM
CLIST EXECUTION
SAMTERM
```

(6) System restart. This function is used to re-initiate scheduling following the execution of the SAMTERM function and will remove the job release inhibitor and initiate a scan of the running matrix.

```
PROC EXECUTION
//STEPNAME EXEC SAMRUNP
//          S      SAMRUNP
CLIST EXECUTION
SAMRUN
```

(7) Posting statistics. This function forces statistics postings from the running matrix to the master data base for any job that has gone through a normal completion, posts date and time, re-computes new average time, and maintains the number of times a job ran. New statistics will be available on any subsequent SASS master data base report (including the percent of time variance of this run as compared to the preceding average time). If a job had previously abnormally ended (but is EOJ now) or was initiated outside of SASS, an exception message is generated, but no statistics are posted.

```
PROC EXECUTION
//STEPNAME SASSPOSTP
```

```
CLIST EXECUTION
SASSPOST
```

(8) Simulation of current workload. This function creates a simulation of currently running jobs and any subsequent jobs to be processed in the scheduling period as defined on the running matrix. (Fig A-15 illustrates this output report.)

```
PROC EXECUTION
//STEPNAME EXEC SASSIME
```

(9) Posting job completion. This function posts whether the job has ended normally (i.e., E) or abnormally (i.e., A). A job is considered to have ended normally if the system completion code equals zero and the user completion code is less than 500. If the job is terminating abnormally, any dependent job is posted with a flush (i.e., F) indicating that it is not releasable by SASS until the abnormally ended job is restarted and completes normally, and if any files were used in update, share, or exclusive-use mode, a lock is placed on that file. After the posting occurs, SASS initiates a scan of the running matrix to determine optimum job(s) to release to the system.

Note. The PROC execution step must be placed as the last jobstep in the job.

(a) Post end to job.

```
PROC EXECUTION
//STEPNAME EXEC SASSENDP
CLIST EXECUTION
SASSEND JN(XXXX)
```

(b) Post ABEND to job.

```
PROC EXECUTION
//STEPNAME EXEC SASSABNDP,JN=JOBNAME
CLIST EXECUTION
SASSABND JN(XXXX)
```

(c) Post end to batch job other than batch job that is executing step.

```
PROC EXECUTION
//STEPNAME EXEC SASSABNDP,ROLL=T,JN=XXXX
```

PROC execution
 //stepname EXEC SAMENDP, JOBNUM=xxx
 //stepname EXEC SAMENDP

Where xxx = The master matrix job number of the job ending. If JOBNUM is not specified, SAM will use the jobname to dynamically determine the correct job number.

CLIST execution
 SAMEND JOBNUM(xxx)
 SAMEND JOBNUM(xxx) FUNKSHN(ABEND)

Where xxx = The master matrix job number of the job ending. The default value of FUNKSHN is END.

(10) System scan. This function initiates a scan of the running matrix to determine optimum job(s) to be released. If the running matrix is termed when this function is executed, then those jobs that are available for release are displayed, but no action is taken.

PROC EXECUTION
 //STEPNAME EXEC SAMSCANP

CLIST EXECUTION
 SAMSCAN

(11) Running matrix update. This function updates the running matrix without interaction with the master data base. The user must verify the resulting running matrix entries, and only data cards are required.

PROC EXECUTION
 //STEPNAME EXEC SAMUPDP
 DATA CARDS

CLIST EXECUTION
 SAMUPD

RESPONSE: ENTER TRANSACTIONS DESIRED
 WHEN THROUGH ENTER /*
 CONTINUATION LINES MUST BE
 ENTERED WITH A BLANK IN COLUMN 1

Data entries are as follows:

(a) Required keywords are as follows:

<u>Keyword</u>	<u>Explanation</u>
★ JN=XXX	Specifies the run matrix job number that is from one to three digits having a value of 1 through 630.
★ FS=RESET	Resets the status and number of users for all run matrix master files. It should not be necessary to execute this option unless an error has occurred in SASS processing. FS is mutually exclusive with other keywords.
★ FN=(LYY,XY,YF)	Change the status of one or more files on the run matrix and automatically adjust the number of users for each file. Each file to be changed must be one to three numeric digits from 1 to 255 and preceded by an L (lock), X (lock except for reference users, or F (free and initiate a scan to reset the status according to the current users of the file). FN used for this purpose is mutually exclusive with other keywords.
★ ND=XXXYY	Change the number of devices assigned to a file on the run matrix where: XXX-file number (three numeric digits from 001 through 255). YY-number of devices (two numeric digits from 01 to 99). Keyword ND is mutually exclusive with other keywords.

(b) Optional keywords to be used with keyword JN are as follows:

<u>Keyword</u>	<u>Explanation</u>
RM=CLEAR	Clears start and end from a job. The job must have started and ended.
★ RM=DELETE	Deletes the job from the run matrix. Job must not have started or already started and ended.
ES=XXXX	Specifies the early start time in HHMM format that is to be the time that the job is to first be considered for scheduling purposes.
★ DP=(XXX,XXX)	Specifies run matrix job numbers that constrain the job being modified. Each must be from one to three numeric digits with a value of 1 through 630. The DP value must be enclosed in parentheses even if one job number or NONE. number or NONE is specified; the maximum number of dependent job numbers is 12 and the default for ADD is NONE.
JM=XXXXXXXX	Specifies JOENAME; required keyword for ADD.

<u>Keyword</u>	<u>Explanation</u>
CR=XXX	Specifies the region required to execute; must be from one to four numeric digits; default for ADD is 54K.
PR=X	Specifies the relative priority that this job should receive for scheduling purposes. If entered, the value must be 15 through 61, default for ADD is 62.
DT=YDDD	Specifies four-digit Julian start date.
T9=XX	Specifies the number of each type of device required for the job; must be numeric; default for ADD is zero.
T7=X	
TD=XX	
T6=XX	
PT=X	
PC=X	
☆ FN=(RXX,UXX,SXX)	Specifies the file number(s) of the master files used by the job. Each number must be one to three numeric digits from 1 to 255 and preceded by an R (reference), U (update), S (share), or E (exclusive). To remove files from the job's run matrix entry, specify FN=(NONE). The default for ADD is NONE.
☆ DN=XXXX	Specifies the master matrix job number of the job being added or changed; must be from one to four numeric digits from 1 through 5760; required keyword for ADD.
CL=X	The security or privacy classification of the job. N = none, default is N R = privacy S = secret U = confidential V = for official use only
☆ CP=X	To identify job as critical path requires Y, default for ADD is N. M = High CPU/non-critical path N = Non-critical/low CPU Y = Critical path/low CPU X = Critical path/high CPU
MD=(XXXXXXXX)	The machine(s) the job must run on, must be eight alphabetic positions (Y or N). If job must run on machine one or three, enter (YNYNNNNN). Entry is required if job is ADD.

HM=(XYXYZZZZ) Assign the timeframe a machine will be unavailable for batch processing. X-represents the machine number, YYYY-represents the starting time, and ZZZZ-represents the ending time. EXAMPLE: HM=(307021700) machine number three will be unavailable for batch processing from 0700 hours until 1700 hours. If the average run time for a new job is two hours and a job ends in machine three at 0515 hours, the new job cannot be run in machine three until after 1700 hours. This may also be used to stop batch processing in order to cycle the machine down. If used to cycle machine down, be sure and remove after restarting processing. If not removed, the machine will cycle down again in 24 hours.

AM=(XXXXXXXX) Application name.

(12) Display jobs on network spool. During the course of a process cycle, the control person needs to know what job or jobs are loaded on the network job spool. If the control person wants to check on a specific job or jobs, he/she needs to enter only those, or enter /* to display all the jobs on the network job spool.

PROC EXECUTION
//STEPNAME EXEC SAMDISP

CLIST EXECUTION
SAMDISP

(13) Cancel a job from network job spool. It may be necessary to cancel a job from the network job spool. Just enter the job name(s) separated by comma(s), one full line only.

PROC EXECUTION
//STEPNAME EXEC SANCANCP
JOBNAME 1, JOBNAME2, JOENAME3, JOENAME4, JOENAME5, JOENAME6, JOENAME7

/*

CLIST EXECUTION
SANCANC

(14) Post ABEND flags to all jobs processing on a machine that is down.
X represents the numeric machine number that is down.

PROC EXECUTION
//STEPNAME EXEC SANDOWNF,SYS=X

CLIST EXECUTION
SANDOWN SYS(X)

(15) Status of a job. During a process cycle, the control person needs to know if a job has been loaded. This procedure will tell the control if a specific job is on the following hold queue, job spool, or permanent JCL file.

```
CLIST EXECUTION
SAMST
JOBNAME1,JOBNAME2,JOBNAME3
/*
```

(16) Activate a job if resources are available, even if system is quiesced. This function will perform the same function as a SAM restart except that only the requested job may be released if resources are available.

```
CLIST EXECUTION
SAMSTART JN(XXXX)
```

(17) Display job names on the EXECUTION JCL file.

```
CLIST EXECUTION
SAMNAME5
```

(18) Delete a job from the EXECUTION JCL file.

```
CLIST EXECUTION
SAMDEL MEMBERNAME
```

(19) List a job on the EXECUTION JCL file.

```
CLIST EXECUTION
SAMLIST MEMBERNAME
```

(20) (a) Copy a member to EXECUTION JCL file from another PDS.

```
CLIST EXECUTION
SAMCOPY OTHERPDSNAME(OTHER PDS MEMBERNAME)
```

(b) Copy a member to PERMANENT JCL from another PDS.

```
CLIST EXECUTION
SAMPCPY WHERE XXX=PDS YYY=MEMBER
```

(21) Load a job on SAM spool from the EXECUTION JCL. SAMCOPY will copy the member in its entirety.

```
CLIST EXECUTION
SAMGENER XXXXXXXX
```

WHERE XXXXXXXX=JOB NAME

(22) List a job on the Network Job Spool.

```
CLIST EXECUTION
SAMSPPOOL XXXXXXXX
```

WHERE XXXXXXXX is the job name that is to be listed.

(23) Load JCL from EXECUTION JCL to network job spool. To be used only when FLY/RUN ABENDs prior to loading JCL and after jobs have been added to the running matrix.

```
//JOBNAME          JOB
//SCRCOPY1         EXEC   SCRCOPYP
//SCRCOPY2         EXEC   SCRCOPYP,COND=(200,LT)
//SCRCOPY.SYSIN    DD     DSN=C.SCRAM.AC70ASB71
//SCRCOPY.CRDSOUT  DD     DSN=C.SCRAM.AC70ASB70
//SCRCOPY3         EXEC   SCRCOPYP,COND=(200,LT)
/*
```

Note. A back-up copy of the above JCL is on the execution JCL file as member name=SCRCOPY.

(24) Temporary change to permanent JCL file. If for any reason a temporary one-time change to a job on the permanent JCL file is desired, execute the following procedure:

```
CLIST EXECUTION
SAMPEDIT XXXXXXXX
```

Where XXXXXXXX is the job name that is to be changed.

When the CRT input lock goes off, edit and change the job.

Note. To request resource requirements to be re-completed, insert a.

```
//*DREAMS SCAN CARD IN FRONT OF THE JOB CARD.
```

```
SAVE
END
DREAMST XXXXXXXX
```

When the CRT returns to ready, the edited job will be on the job spool.

★ (25) To format the running matrix.

```
CLIST execution
SAMFMAT
```

3-6. Operational considerations. a. Required when SCRAM is active. There are two specific instructions that must be adhered to for the efficient operation of SAM.

(1) If the computer system is to be quiesced, always use the SAMTERM function of SAM as opposed to computer operator commands to the computer system. This will prohibit the releasing of jobs to the system by SAM, thus causing the system to dry up.

(2) SAM only considers Class A initiators as available for use.

b. Special conditions.

(1) Releasing a job in the running matrix manually. When the SAMEND function for that job is processed, the running matrix will reflect an E with no S flag and a zero end time and elapsed time. It will not inhibit releasing of dependent jobs and will not be posted by the SAMPOST function to the master matrix.

(2) Modification of normal completion codes. The SAMEND cataloged procedure is internally condition coded, so that user completion codes of greater than 499 are considered to be abnormal. In the event that this requires modification, it could be handled in one of two ways:

(a) PROC modification. If all jobs within the data base are equally affected, the cataloged procedure should be modified accordingly.

(b) PROC overrides. If selected jobs are affected, the JCL required to override must be as follows:

```
//stepname EXEC SAMENDP,COND.END=(xxx...)
```

Where xxx = proper condition codes that reflect the event in which the user wants SAM to post an ABEND to the running matrix.

(3) Running matrix save and recovery procedures. SASSFLYP and SASSRUNP cataloged procedures have an additional step that copies the old running matrix to data set C.WSCRAM before any updates are made. To recover the old running matrix (in case of serious system problems) execute the following JCL:

```
//R.AT EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=C.WSCRAM,DISP=SHR
//SYSUT2 DD DSN=C.RSCRAM,DISP=SHR
//SYSIN DD DUMMY
```

c. Parameter values for the SAMEND program.

(1) The SAMENDM program is executed by the following PROCs (and CLISTS) in the SAM portion of the MMS: SAMENDP (SAMEND), SAMENVP (SAMENV), SAMRESTP (SAMREST), SAMRUNP (SAMRUN), SAMSCANP (SAMSCAN), and SAMTERMP (SAMTERM). There are seven parameters input to SAMEND. Their definition, symbolic name to override, if present, and permissible values are described below.

<u>Definition</u>	<u>Symbolic</u>	<u>Permissible values</u>
Operating mode	Hardcoded	B--batch mode. T--terminal mode.
Action to be performed	Hardcoded	E--post normal job termination. A--post abnormal job termination. I--obtain system resources. RESTART--restart an abnormally ended job. Q--remove the term from running matrix and scan for jobs to release. D--post ABEND to all jobs on a specific machine.

<u>Definition</u>	<u>Symbolic</u>	<u>Permissible values</u>
☆		N--scan running matrix for jobs to release.
☆		T--flag the running matrix to prevent the releasing of jobs.
Master matrix job	JOENUM	Master matrix job number to be processed. It pertains to restarting jobs; if the PROC executed is part of the job being processed by SAMEND, then this parameter is not required because SAMEND will use the JOBNAME to determine the JOENUM.

(2) The hardcoded values and symbolic defaults for each CLIST and PROC are described below.

<u>Parameter</u>	<u>CLIST/PROC</u>	<u>Symbolic default</u>	<u>Hardcoded value</u>
Operating mode	All CLISTs (except those ending in E)		T
	All PROCs (and CLISTs ending in H)		B
Action to be performed	SAMEND SAMENDP	FUNKSHN(FND)	END or ABEND (Depending on condition code.)
	SAMENV(P) SAMREST(P) SAMRUN(P) SAMSCAN(P) SAMTERM(P)		ENV RESTART RUN SCAN TERM
Master data base DSN	All CLISTs		Unknown
	All PROCs		Unknown
☆ Running matrix	All CLISTs All PROCs	RMATRIX(ALMSA.RMATRIX) RMATRIX=ALMSA.RMATRIX	
Master matrix number	SAMEND SAMREST(P)	JOENUM=MONUM (Job number must be specified to execute restart.)	
	SAMENDP		

*3-7. Processing considerations. MMS controls only the job using class A initiators. Prior to the execution of MMS in each CPU, one or more class A initiators must be started.

CHAPTER 4

DYNAMIC RESOURCES EXAMINATION AND MANAGEMENT SYSTEM

4-1. Concept. DREAMS facilitates the scheduling of the MMS network by providing three sources of JCL to all processors within the scheduling network.

a. The network job spool contains jobs stored as sequential data sets. A DREAMS reader reads raw JCL and writes jobs on the network job spool. SAM reads a specific job from the network job spool and releases it to a processor. SAM will delete the job from the network job spool after the job is released.

* b. The permanent JCL file (DSN=C.ALMSA.MMS.SAMJCL and DDNAME = SAM JCL) is a partition data set containing jobs that do not require JCL modification (e.g. parameters, hardcoded volume serials, etc.).

* c. The execution JCL file (DSN=ALMSA.MMS.EXECJCL and DDNAME=SCRMJCL) is a partition data set containing jobs that require JCL modification (e.g. parameters, hardcoded volume serials, etc.). This file is accessed only during the execution of a SASSFLY or SAMEDIT. Once a job has been selected for processing, it is written to the network job spool for initiation as determined by SAM. SAM releases jobs for execution from one of three facilities. The facilities are searched in the following order:

* (1) The OS/PWT or OS/MVS hold queue on the specific processor the job is executed on.

2) The network job spool, which is available to all processors in the MMS network.

(3) The permanent JCL file, which is available to all processors in the MMS network.

Note. If the job is not found on any of the three facilities, the job will not be released.

4-2. DREAMS functions. a. The standard functions unique to DREAMS are to:

- (1) Build the network job spool.
- (2) Build the execution JCL file.
- (3) Eliminate Typerror=Hold.

b. The optional functions unique to DREAMS are to:

- (1) Update the master data base with correct resources.
- (2) Update running matrix with correct resources.
- (3) Edit accounting.
- (4) Change JOBCARD class to A.

4-3. Execution by function. a. General. The execution of various functions of DREAMS provides flexibility of jobstream control in the production environment. The execution of the function under DREAMS is done by PROCs and CLISTs as shown in appendixes C and D. Each function has one or more associated PROCs and CLISTs.

b. Execution options.

(1) Selected PROCs associated with DREAMS processing can be initiated into the system by three methods:

- (a) Through an EXEC card (e.g., //EXEC PROCNAME).
- (b) Through the system reader (e.g., // S PROCNAME).
- (c) Through the system console (e.g., S PROCNAME).

(2) Each CLIST associated with DREAMS is initiated through the terminal (e.g., CLISTNAME) using TSO.

c. Function definition. The definition and performing characteristics of each DREAMS function, and the JCL required, are:

(1) Update the network job spool. This function reads raw JCL and writes jobs to the network job spool. This function require two procedures:

(a) DREAMSR. This procedure scratches any unused data set and allocates 50 data sets on the network job spool pack. DREAMSR automatically starts DREAMS.

(b) DREAMS. This procedure reads JCL and writes jobs to the network job spool. The maximum number of jobs that can be read in one execution of DREAMS is 50. If more than 50 jobs are input to DREAMS in one group, DREAMS will process 50 jobs then display a message on the system console (XXDREAMS ENDING). Prior to the operator restarting DREAMSR, the reader must be cleared and the input card stack reset to resume reading on the 51st job card (this will be the last card read). If more than 35 jobs, but less than 50 jobs are read in one group, DREAMS will process the jobs then display a message on the system console. When DREAMS ends, it automatically starts DREAMSR. When reading directly from the card reader, the end of file must be on. To close DREAMS, read only one slash-slash card with the end of file on the card reader.

Note. Reader unit default is 00C, if another reader is desired, add READER=XXX to the execute statement. If a different media is desired, enter:

```
//DREAMS.RAWJCLIN DD DESIRED - MEDIA - DATE
PROC EXECUTION
//STEPNAME EXEC DREAMSR
//          S      DREAMSR
```

* (c) Options. If requested, DREAMS will update the running matrix with current resource requirements (tape and core). To request matrix updates, insert a // and DREAMS card in front of the job card. DREAMS will edit the accounting number to prevent any manual correction of these currently rejected errors. If this edit is desired, add DD card AC7ABCC1 to the DREAMS PROC. DREAMS will change the job class to A in the job card. Using the SAMCOPY procedure (program APCAOB) if the first parameter is specified as Y the TYPRUN HOLD JCL parameter will not be changed. Otherwise if N, the JCL is changed to NOTIFY-HOLD (useful if scheduling spool option). If the second parameter is specified as N the 'CLASS' parameter will remain as originally specified. Default is Y which will change job card to 'CLASS=A'. When executing program APCASB, which loads jobs from PDS to spool (usually done in SCRCOPY step of SASSFLYP.SASSRUNP), if the first parameter is specified as Y then NOTIFY-HOLD is changed to TYPRUN-HOLD and JCL is written to the internal reader. Otherwise if N, the JCL will be loaded to spool. If the second parameter is specified as N, in this spool option, the CLASS= parameter will remain as originally specified. Default is Y which will change job card 'CLASS=A'.

(2) Update execution JCL file and master data base. This function updates the master data base with current resource requirements (TAPE AND CORE) and adds raw JCL to the EXECUTION JCL file. For first time jobs, or at release time, use SAMCOPY.

PROC EXECUTION

EXECUTE FROM PDS

```
//STEPNAME EXEC DREAMSP,DSN='XXXXXX(MMMMM)'
```

(a) Option. This function updates the master data base with current resource requirements (TAPES and CORE).

PROC EXECUTION

```
//STEPNAME EXEC DREAMSB,DSN='XXXXXX(MMMMM)'
```

Note. The EXECUTION JCL will have a // and /*EOF record added at the end of each job. Updating the EXECUTION JCL file through DREAMS is mandatory. Do NOT use IEBUPDTE or EDIT. If raw JCL is used to load EXECUTION JCL file sequence numbers, the numbers must be in raw JCL.

WARNING! Do not delete member name Z9999999.

```
XXXXXX-DATA SET NAME
XXXXXX-MEMBER NAME
```

(b) To execute procedures DREAMSP or DREAMSB using raw JCL, leave the DSN= off and insert the following:

```
//DREAMS.RAWJCLIN DD DSN=,DISP=,DATA,DLM='??'
RAW JCL
```

(3) Temporary change to execution JCL file. If for any reason, a temporary one-time change to a job on the execution JCL file is desired, execute the following procedure.

CLIST EXECUTION

SAMEDIT XXXXXXXX

WHEN CRT INPUT LOCK GOES OFF, EDIT AND CHANGE JOBS.

Note. To request resource requirements to be re-computed, insert a //*DREAMS SCAN card in front of the job card.

SAVE

END

DREAMST XXXXXXXX

When the CRT returns to ready, the edited job will be on the job spool.

XXXXXXXX-JOB NAME THAT IS TO BE CHANGED.

Appendix A

SASS REPORTS OR OUTPUTS

INSTL = ALMSA

** M M 5 **

PAGE 1

 * MATRIX REPORT *
 *
 * BY JUBNUM *
 * JUL 22 1925 1545

MATRIX = AP, JUBFILE

FUNCTION = REPORT

MATRIX = LNKACN

JUL 22 1925 1545

JOB NUM	JOB NAME	PRI C/D	VOL NUM	REGA (K)	FUNCTIONAL DEPENDENCIES	MASTER FILE UTILIZATION	C PL	T 9	T 7	T U	T 6	T P	RECENT DATE	AVR TIME	RUN TIME	% VAR	PACHIE TIME	ES	
1	F703	1	702	1000	1163,1164,1165	FRET-R	N	N	1	U	U	U	U	U	U	0	0	7	A2ANNNN 3833
2	A433	3	403	132	244,58,387	** NONE **	N	N	2	U	U	U	U	U	U	0	0	100	A2ANNNN
3	G703	1	702	1000	1164,1165,1167,1	** NLNE **	N	N	1	U	U	U	U	U	U	0	0	0	A2ANNNN
4	A717	1	717	512	523,534	** NLNE **	N	N	1	U	U	U	U	U	U	0	0	0	A2ANNNN
5	A717	1	717	512	4	** NLNE **	N	N	1	U	U	U	U	U	U	0	0	0	A2ANNNN
6	A702	1	702	512	111,910	WRL-U,FMJ-R	N	N	1	U	U	U	U	U	U	0	0	0	A2ANNNN
7	F703	1	703	512	3	** NLNE **	N	N	0	U	U	U	U	U	U	0	0	0	A2ANNNN
8	A445	2	445	128	** ACNE **	MFFUF-J,IFPCF-U	N	N	0	U	U	U	U	U	U	0	0	0	A2ANNNN
9	A479	1	475	54	2	** NONE **	N	N	0	U	U	U	U	U	U	0	0	0	A2ANNNN
10	A824	1	824	650	** NONE **	APAC-R,AAU-R	N	N	1	U	U	U	U	U	U	0	0	0	A2ANNNN
11	A824	1	824	152	10	** NLNE **	N	N	2	U	U	U	U	U	U	0	0	0	A2ANNNN
12	B530	2	530	140	67	LEPR-R,PFPR-R	N	N	0	U	U	U	U	U	U	0	0	0	A2ANNNN
13	RDE615	1	615	512	508	** NLNE **	N	N	0	U	U	U	U	U	U	0	0	0	A2ANNNN
14	A597	1	597	175	** ACNE **	MFFSN-R	N	N	1	U	U	U	U	U	U	0	0	0	A2ANNNN
15	A632	2	632	250	9100,9322	IFFSN-R,MFFSN-R	N	N	6	U	U	U	U	U	U	0	0	0	A2ANNNN
16	B471	1	471	300	971	MFABL-U,IFABL-U	N	N	0	U	U	U	U	U	U	0	0	0	A2ANNNN
17	C884	1	884	512	1121	** NLNE **	N	N	1	U	U	U	U	U	U	0	0	0	A2ANNNN
18	E404	2	404	250	4672	** ACNE **	N	N	1	U	U	U	U	U	U	0	0	0	A2ANNNN
19	B469	2	469	250	235	IFFSN-R,MFFSN-U, MFFIA-U,IFFIA-U	N	N	1	U	U	U	U	U	U	0	0	0	A2ANNNN
20	A715	1	715	850	602,182	EAAX-U,IFPXR-R, MFPXR-R,MFPCR-R	N	N	1	U	U	U	U	U	U	0	0	0	A2ANNNN
21	A432	2	432	230	537	MFFSN-R	N	N	2	U	U	U	U	U	U	0	0	0	A2ANNNN
22	A517	1	517	200	** NONE **	IFFSN-R,MFFSN-U	N	N	1	U	U	U	U	U	U	0	0	0	A2ANNNN
23	B715	1	715	450	2,20	EAAX-U,IFPXR-R, MFPXR-R,MFPCR-R	N	N	1	U	U	U	U	U	U	0	0	0	A2ANNNN
24	C715	1	715	512	20,23	** NLNE **	N	N	2	U	U	U	U	U	U	0	0	0	A2ANNNN
25	A496	1	496	780	** NONE **	DRS-E	N	N	0	U	U	U	U	U	U	0	0	0	A2ANNNN
26	A620	2	620	100	2,400	IFFSN-R	N	N	0	U	U	U	U	U	U	0	0	0	A2ANNNN
27	B522	2	522	260	** ACNE **	MFFSN-R	N	N	2	U	U	U	U	U	U	0	0	0	A2ANNNN
28	B496	1	496	820	25	DRSH-E	N	N	1	U	U	U	U	U	U	0	0	0	A2ANNNN
29	B587	1	587	512	248,355,357,358, 359,395,396	IFFSN-R,MFFSN-R	N	N	0	U	U	U	U	U	U	0	0	0	A2ANNNN
30	A602	1	602	150	125	IFFSN-R,MFFSN-U	N	N	3	U	U	U	U	U	U	0	0	0	A2ANNNN
31	B602	1	602	160	30		N	N	0	U	U	U	U	U	U	0	0	0	A2ANNNN
32	C602	1	602	150	31		N	N	1	U	U	U	U	U	U	0	0	0	A2ANNNN
33	D602	1	602	160	32		N	N	0	U	U	U	U	U	U	0	0	0	A2ANNNN
34	A513	2	513	512	547,84008	** NLNE **	N	N	1	U	U	U	U	U	U	0	0	0	A2ANNNN
35	A634	2	634	300	150,170,153,137, 157,469,35	IFPAU-U,MFMAD-U	N	N	4	U	U	U	U	U	U	0	0	0	A2ANNNN
36	F602	1	602	160	33		N	N	0	U	U	U	U	U	U	0	0	0	A2ANNNN
37	B513	2	513	512	34	IFFSN-R,MFFSN-R	N	N	1	U	U	U	U	U	U	0	0	0	A2ANNNN
38	C513	2	513	512	34,37	** ACNE **	N	N	1	U	U	U	U	U	U	0	0	0	A2ANNNN

Appendix A--Continued

INSTL = ALMSA

00 A M S 00

PAGE 2

MMATHIX = AP-JCSFILE

FUNCTION = REPORT

MMATHIX = LARALMA

JUL 24 1985 1345

VIN	NAMF	VOL	NAME	VCL	NAME	VCL	NAME	VOL	NAME	VCL	NAME
0257		0289		0321		0353		0417	RUVCU	0445	PPFNURO
0258		0290		0322		0354		0418	PREJCU	0450	CCD
0259		0291		0323		0355		0419	SCSCLAN	0451	FSANLY
0260		0292		0324		0356		0420	SCSLEV	0452	FSANKEY
0261		0293		0325		0357		0421	SCSPKT	0453	PAOVED
0262		0294		0326		0358		0422	SCITENS	0454	FLAUPRO
0263		0295		0327		0359		0423	SCNSURY	0455	SCUPCCN
0264		0296		0328		0360		0424	AMDFUPJ	0456	PLMREAL
0265		0297		0329		0361		0425	AMDFLNG	0457	SACUPOT
0266		0298		0330		0362		0426	SCAPPO	0458	LSANFY
0267		0299		0331		0363		0427	MTHPNU	0459	PPFPMT
0268		0300		0332		0364		0428	AUTGDIS	0460	RFMILCA
0269		0301		0333		0365		0429	PUSAND	0461	2PUBREX
0270		0302		0334		0366		0430	EIPUPDI	0462	FSNBLDG
0271		0303		0335		0367		0431	FINMUEN	0463	ILLASE
0272		0304		0336		0368		0432	SIMXSEL	0464	ICFINQ
0273		0305		0337		0369		0433	CPSUPDI	0465	ILLNG
0274		0306		0338		0370		0434	STDRUPJ	0466	FLUPUDI
0275		0307		0339		0371		0435	BURFCUN	0467	FFCHANI
0276		0308		0340		0372		0436	BURLOWF	0468	ALTUOSP
0277		0309		0341		0373		0437	ILSUPPA	0469	PLKASA
0278		0310		0342		0374		0438	VRSEUIT	0470	
0279		0311		0343		0375		0439	FINUP	0471	SAPL
0280		0312		0344		0376		0440	PPORAUT	0472	ANUSIP
0281		0313		0345		0377		0441	JCUEAPL	0473	FINMCPN
0282		0314		0346		0378		0442	BUGGSTR	0474	LASE
0283		0315		0347		0379		0443	PHYINV	0475	BASS
0284		0316		0348		0380		0444		0476	
0285		0317		0349		0381		0445	ANSLIDE	0477	GRUSS
0286		0318		0350		0382		0446	MASSLAN	0478	FINCCP
0287		0319		0351		0383		0447	PLUPUDI	0479	MASLIMS
0288		0320		0352		0384		0448	AMUPUDI	0480	TRFP

PAGE 1

FUNCTIO = REPLA

JUL 22 1965 1545

A-3

Figure A-3. Application report by application name

Appendix A--Continued

INSTL = ALMSA

1304

FUNCTION = REPORT

5451 27 709

MMATRIK = AP. JCBFILE

K MATRIX = LANCZOS

[illegible]

INSTL = ALMSA

PAGE 1

MMATRIX = AP.JCBFILE

FUNCTION = REPORT

MMATRIX = LARACON

JUL 22 1965 1345

NAME	NUM	TYPE	NAME	NUM	TYPE	NAME	NUM	TYPE	NAME	NUM	TYPE
AAU	103	S	IFDMS	113	M	IFPAU	20	P	IFRSA	73	M
APAC	102	S	IFELP	30	M	IFPAU	22	P	IFSAU	3	M
APAGM	104	S	IFFAU	11	M	IFPAU	56	P	IFSPA	42	M
APARE	101	S	IFPAR	52	M	IFMLE	24	P	IFRS	5	A
CJIF	74	M	IFPCS	47	M	IFPFI	92	P	IFLUP	39	M
DMS	119	S	IFFCT	13	M	IFPMH	32	P	IFTAI	10	M
DMSH	120	S	IFFEC	14	M	IFCCD	25	P	LJAR	58	S
FAAX	50	S	IFFGL	16	M	IFPDF	27	P	MCNO	104	S
FAMJ	19	O	IFFIA	59	M	IFPEM	66	P	PLUTE	111	S
FHFT	118	S	IFFIS	17	M	IFPHM	54	P	MFABL	46	M
IFARL	47	M	IFFSA	43	M	IFPPE	34	P	MFCLF	61	M
IFACP	40	M	IFFSC	33	M	IFPLF	63	P	MFECF	7	M
IFCPC	1	M	IEESH	3	M	IFPRR	36	P	MFENS	114	M
IFOCF	62	M	IFFUR	51	M	IFRSF	64	P	MFELP	31	M
IFOCM	29	M	IFIDL	2	M						

THE FOLLOWING FILE NUMBERS ARE UNASSIGNED --

075, 076, 077, 078, 079, 080, 081, 082, 083, 084, 085, 086, 087, 088, 089, 090, 091, 092, 093, 098, 099, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255

Appendix A--Continued

```

INSTL = ALMSA
*****
*
* SYSTEM
*
* MES LINES
*
*****
FUNCTION = REPORT
*****
JUL 22 1965 1345
*****

```

AVAILABLE CCKE JLVICEJ -- BVFL4U E -- UJ4NY UC3J

40404

DISA

22

22

2

2

5

5

2

2

2

2

—

—

Appendix Continued

INSTL = ALMSA
 MATRIX = AP-JCBFILE
 MATRIX = UNKNOWN

 J C H E C U L I N G
 N E T C L E R

FUNCTION = REQUEST
 JUL 22 1965 1945

MACH1 MACH2 MACH3 MACH4 MACH5 MACH6 MACH7 MACH8

MACHINE ID A2 A1 A3 A4

NUM-AVAILABLE START TIME 0700

NUM-AVAILABLE END TIME 1630

TAPED USED AS TAPE9 N Y N N

TAPF6250 USED AS TAPE9 N Y N N

Appendix A--Continued

PAGE 1

INSTL = ALPSA

MMATRIX = AP.JC6FILE

FUNCTION = REPORT

RMATRIX = CARMLNA

JUL 24 1985 1545

APPI NAME APPI VUL AVG TIME JOB NAME

AMMNT	803	3	AFC3.DD03	
ARMDFI	498	10	A458	
ARFBI	884	0	CE64.A884.8884	
ATKMT	540	5	A540	
ATKMT	608	0	A608	
ATKMT	425	40	A425.D425.C425.D425.E425.F425.G425	
ATKMT	505	4	A505	
ATKMT	424	52	A424.D424.C424.E424.F424.G424.H424.I424	
ATKMT	576	73	A576.D576.E576.F576.G576.H576.I576	
ATKMT	445	7	A445	
ATKMT	817	13	A817.D817.E817.F817.G817.H817.I817	
ATKMT	813	0	MCNE **	
ATKMT	825	0	A825.D825.C825.E825.F825.G825.H825.I825	
ATKMT	826	0	A826.E826.F826.G826.H826.I826	
ATKMT	818	32	A818	
ATKMT	820	5	7820.A820	
ATKMT	819	76	A819.D819.E819.F819.G819.H819.I819	
ATKMT	811	11	A811.D811.E811.F811.G811.H811.I811	
ATKMT	809	14	A809.D809.E809.F809.G809.H809.I809	
ATKMT	821	6	A821.D821.C821.E821.F821.G821.H821.I821	
ATKMT	810	5	A810.D810.E810.F810.G810.H810.I810	
ATKMT	812	5	A812.D812.E812.F812.G812.H812.I812	
ATKMT	814	2	A814	
ATKMT	816	5	A816.D816.E816.F816.G816.H816.I816	
ATKMT	801	16	A801.D801.C801.E801.F801.G801.H801.I801	
ATKMT	805	43	A805.D805.C805.E805.F805.G805.H805.I805	
ATKMT	824	0	A824.D824.E824.F824.G824.H824.I824	
ATKMT	815	15	A815.D815.E815.F815.G815.H815.I815	
ATKMT	807	33	A807.D807.E807.F807.G807.H807.I807	
ATKMT	804	5	A804.D804.E804.F804.G804.H804.I804	
ATKMT	806	174	A806.D806.C806.E806.F806.G806.H806.I806	
ATKMT	801	0	MCNE **	
ATKMT	808	50	A808.D808.C808.E808.F808.G808.H808.I808	
ATKMT	802	6	A802.D802.E802.F802.G802.H802.I802	
ATKMT	544	29	A544.D544.E544.F544.G544.H544.I544	
ATKMT	472	69	A472.D472.E472.F472.G472.H472.I472	
ATKMT	523	20	A523	
ATKMT	542	0	MCNE **	
ATKMT	428	21	A428.D428.C428.E428.F428.G428.H428.I428	
ATKMT	468	12	A468.D468.E468.F468.G468.H468.I468	
ATKMT	822	13	A822.D822.E822.F822.G822.H822.I822	
ATKMT	543	12	A543	
ATKMT	475	63	A475.D475.E475.F475.G475.H475.I475	
ATKMT	519	24	A519.D519.E519.F519.G519.H519.I519	

Q MATRIZ = LAKALAKA

JUL 22 1985 1345

INTHA-APPLICATION JCB DÉPENDENCIES

140/3815R
640/664V
** ENCA **
** C570057
** ENDN **
040/3330
** ENCA **
** 50/5050
** ENCA **
1RC/418V

[illegible]

000 NONE 00
000 VIVE 00
000 MVE 00
000 NVE 00
000 NONE 00
000 NONE 00
000 NONE 00
000 MVE 00
000 NONE 00
000 NONE 00
000 NONE 00

1100/291000

NAME **DATE** **SCORE**
NAME **DATE** **SCORE**
NAME **DATE** **SCORE**
NAME **DATE** **SCORE**
NAME **DATE** **SCORE**
8803/080

Recommendation

Appendix A--Continued

UNSTL - ALMSA S C R A M PAGE 1

.....

MATRIX - C.NSCRAM P R O C E S S I N G FUNCTION - FLY

MATRIX - C.ZSCRAM L O O MAR 31 1980 1504

.....

----- ON-THE-FLY PROCESSING MESSAGES -----

1988-C JOB 'NR1 PROCESSED TO EOJ

3182-C POSTING HAS OCCURRED

3170-C DELETED FROM RMATRIX

1988-C JOB 'NR5 PROCESSED TO EOJ

3182-C POSTING HAS OCCURRED

3170-C DELETED FROM RMATRIX

Appendix A--Continued

INSTR	ALMSA	SE	SCRAM	DE	PAGE
1	*****				3
2	*****				
3	*****				
4	*****				
5	*****				
6	*****				
7	*****				
8	*****				
9	*****				
10	*****				
11	*****				
12	*****				
13	*****				
14	*****				
15	*****				
16	*****				
17	*****				
18	*****				
19	*****				
20	*****				
21	*****				
22	*****				
23	*****				
24	*****				
25	*****				
26	*****				
27	*****				
28	*****				
29	*****				
30	*****				
31	*****				
32	*****				
33	*****				
34	*****				
35	*****				
36	*****				
37	*****				
38	*****				
39	*****				
40	*****				
41	*****				
42	*****				
43	*****				
44	*****				
45	*****				
46	*****				
47	*****				
48	*****				
49	*****				
50	*****				
51	*****				
52	*****				
53	*****				
54	*****				
55	*****				
56	*****				
57	*****				
58	*****				
59	*****				
60	*****				
61	*****				
62	*****				
63	*****				
64	*****				
65	*****				
66	*****				
67	*****				
68	*****				
69	*****				
70	*****				
71	*****				
72	*****				
73	*****				
74	*****				
75	*****				
76	*****				
77	*****				
78	*****				
79	*****				
80	*****				
81	*****				
82	*****				
83	*****				
84	*****				
85	*****				
86	*****				
87	*****				
88	*****				
89	*****				
90	*****				
91	*****				
92	*****				
93	*****				
94	*****				
95	*****				
96	*****				
97	*****				
98	*****				
99	*****				
100	*****				

FIGURE A-12. INPUT STATEMENTS TO SASSCHK OR SASSFLY

FUNCTION - FLY
MAR 31 1980 1604

JOBS TO BE RUN THRU

SCRAM SIMULATOR

PH JN	PH JN	JOB NAME	PRI CD	VOL NUM	CORE SIZE	AVE TIME	C C P L	T T S 7	T T D 6	T T P 5	ES TIME	MASTER FILE UTILIZATION	MATRIX JOB DEPENDENCIES	MACHINE DEP
1	9	NR9	16	001	111	199	N S	9	1	3	1	1	1	1234NNNN
2	1	NR1	1	001	100	3688	N N	9	1	3	1	1	4 1 6 10 12	12NN66NN
3	4	NR4	1	001	94	98	N N	0	0	0	0	0	11	1NNNNNN
4	5	NR5	9	001	112	45	N N	9	0	0	1	0	0	12NNNN7N
5	10	NR10	11	001	64	130	N R	6	0	0	0	0	0	1NNNNNNN
6	11	NR11	1	001	112	0	Y N	0	0	0	0	0	0	123NNNN7N
7	12	NR12	1	001	64	96	N N	0	1	0	0	0	0	1NNNNNN8
8	13	NR13	1	001	64	110	Y N	0	0	1	0	0	0	1NNNNNNN
9	14	NR14	1	001	64	98	N N	0	0	0	1	0	0	12NNNN678
10	15	NR15	1	001	64	138	Y N	0	0	0	1	0	0	12NNNN178
11	16	NR16	1	001	64	99	N N	6	0	0	0	0	0	1NNNNNNN
12	3625	NR3625	1	001	64	10	N N	0	0	0	0	0	0	11

FIGURE A-13. JOBS TO BE RUN THRU SASS SIMULATION.

Appendix A--Continued

INSTL = ALMSA S C R A M PAGE 1

PMATRIX = C.MSCRAM INTRA-INTER APPLICATION FUNCTION = ELY

RMATRIX = C.ZSCRAM JOB DEPENDENCIES MAR 31 1980 1504

APPL NAME	APPL VOL	JOB NAME	INTRA-APPLICATION JOB DEPENDENCIES	INTER-APPLICATION JOB DEPENDENCIES WITH APPLICATION VOLUME NUMBER
C 001	001	NR1	NR5, NR9, NR14, NR15, NR3525	NR5 NONE **
		NR4	NR5 NONE **	NR4 NONE **
		NR5	NR5 NONE **	NR5 NONE **
		NR9	NR5 NONE **	NR9 NONE **
		NR10	NR13	NR10 NONE **
		NR11	NR10	NR11 NONE **
		NR12	NR10	NR12 NONE **
		NR13	NR5 NONE **	NR13 NONE **
		NR14	NR10	NR14 NONE **
		NR15	NR10	NR15 NONE **
		NR16	NR10	NR16 NONE **
		NR3525	NR16	NR3525 NONE **

A-14

FIGURE A-14. INTER-INTRA APPLICATION JOB DEPENDENCIES.

INSTL = ALM5A

**

M

**

FUNCTION = SIM

MATRIX = ALM5A.MATRIX

RUN

JAN 24 1985 0900

MATRIX = UNKNOWN

SIMULATION

MACHINE = ALL

SYSTEM RESOURCES
FOR SIMULATIONCORE = 4000 INITS = 8 DISK = 48
TAPE9 = 22 TAPE7 = 2 TAPED = 0

TAPE6 = 0 PRINT = 2 PUNCH = 1

TYPE	INIT1	INIT2	INIT3	INIT4	INIT5	INIT6	INIT7	INIT8	DEVICES IN USE									
090	A4.3																	
095																		
097																		
098																		
099																		
100																		
101																		
102																		
103																		
104																		
105																		
106																		
107																		
108																		
109																		
110																		
111																		
112																		
113	A4.2	A5.2	E4.3															
114	K4.2		E4.2															
115	C4.2																	
116	E4.2																	
117	G4.2	F4.2																
118	H4.2	L4.2																
119	I4.2																	
120	J4.2																	
121	K4.2																	

***** PROCESS TIME = 2 HOURS 47 MINUTES *****

Figure A-15. Run simulation

W547V - 7A5R.1

1. **1**

FUNCTION - SIM
JAN 24 1985 0900

IMATRIK - ALISA, IMATRIX

R4412IX - UNKNOWN

**H
U
R**

STIMULATION

[illegible]

0 MINUTES	0000
17 MINUTES	0000
9 MINUTES	0000

	0 HOURS	2 HOURS	3 HOURS
1	100%	98%	97%
2	100%	96%	95%
3	100%	94%	93%
4	100%	92%	91%
5	100%	90%	89%
6	100%	88%	87%
7	100%	86%	85%
8	100%	84%	83%
9	100%	82%	81%
10	100%	80%	79%
11	100%	78%	77%
12	100%	76%	75%
13	100%	74%	73%
14	100%	72%	71%
15	100%	70%	69%
16	100%	68%	67%
17	100%	66%	65%
18	100%	64%	63%
19	100%	62%	61%
20	100%	60%	59%
21	100%	58%	57%
22	100%	56%	55%
23	100%	54%	53%
24	100%	52%	51%
25	100%	50%	49%
26	100%	48%	47%
27	100%	46%	45%
28	100%	44%	43%
29	100%	42%	41%
30	100%	40%	39%
31	100%	38%	37%
32	100%	36%	35%
33	100%	34%	33%
34	100%	32%	31%
35	100%	30%	29%
36	100%	28%	27%
37	100%	26%	25%
38	100%	24%	23%
39	100%	22%	21%
40	100%	20%	19%
41	100%	18%	17%
42	100%	16%	15%
43	100%	14%	13%
44	100%	12%	11%
45	100%	10%	9%
46	100%	8%	7%
47	100%	6%	5%
48	100%	4%	3%
49	100%	2%	1%
50	100%	0%	0%

SEQUENTIAL TIME:

*****MULTIPLYING FACTOR 1.24*****

DEVICES IN USE	
Y	↑
9	7
0	6
1	5
2	4
3	3
4	2
5	1
6	0
7	9
8	8
9	7
0	6
1	5
2	4
3	3
4	2
5	1
6	0
7	9
8	8
9	7
0	6
1	5
2	4
3	3
4	2
5	1
6	0
7	9
8	8
9	7
0	6
1	5
2	4
3	3
4	2
5	1
6	0
7	9
8	8
9	7
0	6
1	5
2	4
3	3
4	2
5	1
6	0
7	9
8	8
9	7
0	6
1	5
2	4
3	3
4	2
5	1
6	0
7	9
8	8
9	7
0	6
1	5
2	4
3	3
4	2
5	1
6	0
7	9
8	8
9	7
0	6
1	5
2	4
3	3
4	2
5	1
6	0
7	9
8	8
9	7
0	6
1	5
2	4
3	3
4	2
5	1
6	0
7	9
8	8
9	7
0	6
1	5
2	4
3	3
4	2
5	1
6	0
7	9
8	8
9	7
0	6
1	5
2	4
3	3
4	2
5	1
6	0
7	9
8	8
9	7
0	6
1	5
2	4
3	3
4	2
5	1
6	0
7	9
8	8
9	7
0	6
1	5
2	4
3	3
4	2
5	1
6	0
7	9
8	8
9	7
0	6
1	5
2	4
3	3
4	2
5	1
6	0
7	9
8	8
9	7
0	6
1	5
2	4
3	3
4	2
5	1
6	0
7	9
8	8
9	7
0	6
1	5
2	4
3	3
4	2
5	1
6	0
7	9
8	8
9	7
0	6
1	5
2	4
3	3
4	2
5	1
6	0
7	9
8	8
9	7
0	6
1	5
2	4
3	3
4	2
5	1
6	0
7	9
8	8
9	7
0	6
1	5
2	4
3	3
4	2
5	1
6	0
7	9
8	8
9	7
0	6
1</	

Appendix A--Continued

INSTL - ALMSA

PAGE 1

 MATRIX - C. MCGRAW
 CROSS REFERENCE BY JOBNAM

 MATRIX - UNKNOWN
 JOB DEPS AND DEP JOBS

 FUNCTION - JMXREF
 MAR 24 1980 1228

JOB NAME	JOB NUM	APPL NAME	VOL	JOB DEPENDENCIES	JOBNAME/JOBNUM/APPLNO	DEPENDENT JOBS
CHG3	3	NR333	333	NR3525/3525/001	NR10/10/001	NR1/1/001
NR1	1	CHG1	001	NR2/2/229 CHG3/3/333 NR5/5/001 NR6/6/200 NR7/7/300 NR8/8/400 NR9/9/001 NR10/10/001 NR13/13/001 NR14/14/001 NR15/15/001 NR3525/3525/001	NR10/10/001	NR1/1/001
NR10	10	CHG1	001	NR9/9/001	NR10/10/001	NR1/1/001
NR11	11	CHG1	001	NR10/10/001	NR10/10/001	NR1/1/001
NR12	12	CHG1	001	NR10/10/001	NR10/10/001	NR1/1/001
NR13	13	CHG1	001	NR10/10/001	NR10/10/001	NR1/1/001
NR14	14	CHG1	001	NR10/10/001	NR10/10/001	NR1/1/001
NR15	15	CHG1	001	NR10/10/001	NR10/10/001	NR1/1/001
NR16	16	CHG1	001	NR10/10/001	NR10/10/001	NR1/1/001
NR2	2	NR999	999	NR10/10/001	NR10/10/001	NR1/1/001
NR3525	3525	CHG1	001	NR10/10/001	NR10/10/001	NR1/1/001
NR4	4	CHG1	001	NR10/10/001	NR10/10/001	NR1/1/001
NR5	5	CHG1	001	NR10/10/001	NR10/10/001	NR1/1/001
NR6	6	NR200	200	NR10/10/001	NR10/10/001	NR1/1/001
NR7	7	NR300	300	NR10/10/001	NR10/10/001	NR1/1/001
NR8	8	NR400	400	NR10/10/001	NR10/10/001	NR1/1/001
NR9	9	CHG1	001	NR10/10/001	NR10/10/001	NR1/1/001

FIGURE A-16. CROSS REFERENCE BY JOB NAME.

HEATFIX - C. MACRAM

CROSS REFERENCE BY APPLICANT

FUNCTION - AMYDEE

RMATRIX • UNKNOWN

JOB DEPS AND DEP JOBS

MAR 24 1980 1328

[illegible]

FIGURE A-17. CROSS REFERENCE BY APPLICATION NAME.

Appendix A--Continued

INSTL - ALMSA

S C R A M

PAGE 1

 MATRIX - C.HSCRAM
 R MATRIX - UNKNOWN

CROSS REFERENCE
 BY FILE ACRONYM

 FUNCTION - FAXREF
 MAR 24 1980 1229

 FILE NAME
 FILE NUMBER

 UPDATE/SHARE/EXCLUSIVE-USE
 JOB NAME/JOB NUMBER

NR1 001 NR1/1 NR4/4 NR5/5

NR2 002 CH03/3 NR4/4

NR3 003 CH03/3 NR4/4

-100 100 NR5/5

-200 200 ** NONE **

-204 204 NR1/1

-255 255 CH03/3 NR5/5

A-20

FIGURE A-19. CROSS REFERENCE BY FILE ACRONYM.

JUL 24 1963 : 1950

MACHINE • ALL

II. I. I. I. I.

RM	MM	JN	NAME	PR	S	E	A	DATE	TIME	TIME	ES	EL	AVG	A	C	VOL	P	L	DEPENDENCIES	RCN	9	7	0	6	7	C	FILES	USED	MACR.	DEP.
1	1	F703	01																											
2	718	A902	01																											
3	719	B902	01																											
4	720	C902	01																											
5	721	D902	01																											
6	726	E902	01																											
7	730	F902	01																											
8	731	G902	01																											
9	732	H902	01																											
10	733	J902	01																											
11	735	K902	01																											
12	735	L902	01																											
13	736	M902	01																											
14	737	N902	01																											
15	738	P902	01																											
16	739	F902	01																											
17	108	A906	03																											
18	111	B906	03																											
19	913	C906	02																											
20	1001	D906	01																											
21	1017	E906	01																											
22	1022	F906	01																											
23	1031	G906	01																											
24	1045	H906	01																											
25	1101	J906	01																											
26	352	K905	02																											
27	423	L905	02																											
28	424	C975	02																											

3974
PAGE 3

75-70 - 71541

44-38861-1000

FUNCTION - ANALYSIS

JUL 24 1995 0757

FILE NUMBER	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	
FILE NAME	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
STATUS	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
SCHMIDT	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
REF-USER	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
AT-REF-USER	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
DENIED-USER	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
RECHIEF-IC	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40

FILE-NAME	59	50	61	62	1-7	1-8
FILE NAME	TYPE	AREA	PRICE	PRICE	PRICE	PRICE
STATS	1	1	3	2	2	1
SCHM USERS	3	0	0	0	0	0
REF USERS	3	0	0	0	0	0
O/T REF USERS	3	0	0	0	0	0
APPENDED JOBS	3	0	0	0	0	0
MOCHTIME ID.						

NAME	ADDRESS	PHONE	TELETYPE	TELEX	MAILING ADDRESS	TELEPHONE	TELETYPE	TELEX
SYSTEM								
NETWORK								

311:667.1 - K131500

631545

FUNCTION = RPTALL

WATSON - 10.2MA 157K

MEMORANDUM

JUL 24 1985 0757

GI 3N11344.

44

13

11

11

14-00000

NON-CLASSIFIED

2113 Jett Cove

11

—

7

1

0-871 37770 35276071

21

4

1

1

Appendix B—Continued

85470 = 715M

— 10 —

PAGE 1

3713967-01 - 6191VMM

RMASCY - 47-69472X

241577

537

FUNCTION = PARTIAL

JUL 24 1985 3:75 PM

GENERAL DISCUSSION

5556-C TOTAL NUMBER OF JAPS INTERVIEWED TO DATE =

2124-C S/A PROCESSING WAS COMPLETED NORMALLY

Appendix B- inued

INST. - ALMSA

-- 3 A M --

PAGE 4

MATRIX - C.MSCRAM

R.M.A.T.R.I.X

FUNCTION - RPIALL

RMATRIX - C.RSCRAM

REPORT

MAR 24 1980 1220

DRIVE=3A0	TYPE=9	OFF	ACCESSABLE=NNNNNNNN
DRIVE=3A1	TYPE=9	OFF	ACCESSABLE=NNNNNNNN
DRIVE=3A2	TYPE=9	OFF	ACCESSABLE=NNNNNNNN
DRIVE=3A3	TYPE=9	OFF	ACCESSABLE=NNNNNNNN
DRIVE=3A4	TYPE=9	OFF	ACCESSABLE=NNNNNNNN
DRIVE=3A5	TYPE=9	OFF	ACCESSABLE=NNNNNNNN
DRIVE=3A6	TYPE=9	OFF	ACCESSABLE=NNNNNNNN
DRIVE=3A7	TYPE=9	OFF	ACCESSABLE=NNNNNNNN
DRIVE=3A8	TYPE=9	DOWN	ACCESSABLE=NNNNNNNN
DRIVE=3A9	TYPE=9	DOWN	ACCESSABLE=NNNNNNNN
DRIVE=3AA	TYPE=9	DOWN	ACCESSABLE=NNNNNNNN
DRIVE=3AB	TYPE=9	DOWN	ACCESSABLE=NNNNNNNN
DRIVE=3AC	TYPE=9	DOWN	ACCESSABLE=NNNNNNNN
DRIVE=3AD	TYPE=9	DOWN	ACCESSABLE=NNNNNNNN
DRIVE=3AE	TYPE=9	DOWN	ACCESSABLE=NNNNNNNN
DRIVE=3AF	TYPE=9	DOWN	ACCESSABLE=NNNNNNNN
DRIVE=3B0	TYPE=9	DOWN	ACCESSABLE=NNNNNNNN
DRIVE=3B1	TYPE=9	DOWN	ACCESSABLE=NNNNNNNN
DRIVE=3B2	TYPE=9	DOWN	ACCESSABLE=NNNNNNNN
DRIVE=3B3	TYPE=9	DOWN	ACCESSABLE=NNNNNNNN
DRIVE=3B4	TYPE=9	DOWN	ACCESSABLE=NNNNNNNN
DRIVE=3B5	TYPE=9	DOWN	ACCESSABLE=NNNNNNNN
DRIVE=3B6	TYPE=9	DOWN	ACCESSABLE=NNNNNNNN
DRIVE=3B7	TYPE=9	DOWN	ACCESSABLE=NNNNNNNN

FIGURE D-1 -- Continued

Appendix C

LIST OF MMS COMMAND PROCEDURES

<u>Paragraph</u>	<u>CLIST</u>	<u>Description</u>	<u>Example</u>
3-5c(3)(b)	SAMABEND	POST ABEND FLAG ON RUN MATRIX A JOB.	SAMABEND JN(9999) 9999 IS JOBNUM
3-5c(13)	SAMCANC	CANCELS/DELETES JOBS FROM JOBSPOOL PACK.	SAMCANC XXXX XXXX=JOBNAME
3-5c(2E)	SAMCOPY	COPY MEMBER FROM A PDS TO EXECUTION JCL (DSN=C.SCRAM. EXEC JCL).	SAMCOPY XXX(444) XXX=PDS YYY=MEMBER
3-5c(18)	SAMDEL	DELETE JOB FROM EXECUTION JCL.	SAMDEL XXXX XXX=MEMBER NAMES
3-5c(12)	SAMDISP/ SAMO	DISPLAY JOBS SPECIFIC JOBS ON THE JOBSPOOL.	SAMDISP ENTER JOBNAMES AFTER PROMPT JOB1, JOB2
4-3c(3)	SAMEDIT	1. TEMP. UPDATE TO EXECUTION JCL (DSN=C.SCRAM.EXEC JCL). 2. FOLLOWED BY DREAMST WHICH EDITS AND PUTS IT OUT ON THE SPOOL.	SAMEDIT XXXX DREAMST XXXX XXX=MEMBER NAME
3-5c(3)(a)	SAMEND	POST END TO RUNNING MATRIX.	SAMEND JN(9999) 9999=JOBNUMBER ON MASTER MATRIX
3-5c(2)	SAMENV	DISPLAY CURRENT SYSTEM ENVIRONMENT.	SAMENV
3-5c(21)	SAMGENER	COPY MEMBER FROM EXECJCL TO SPOOL PACK/TO HOLD QUEUE.	SAMGENER XXX=JOBNAME
3-5c(19)	SAMLIST	LIST JOB ON EXEC JCL.	SAMLIST XXX XXXX=MEMBER NAME
3-5c(17)	SAMNAMES	DISPLAY ALL JOBNAMES ON EXECUTION JCL.	SAMNAMES
* 3-5c(20)(a)	SAMCOPY	COPY MEMBER FROM PDS TO EXECUTION JCL.	SAMCOPY
3-5c(2F)(E)	SAMPCPY ¹	COPY MEMBER FROM A PDS TO PERMANENT JCL (DSN=C.SAMJCL).	SAMPCPY XXX(444) XXX=PDS YYY=MEMBER

¹These command procedures are not previously documented in this publication.

Appendix C--Continued

<u>Paragraph</u>	<u>CLIST</u>	<u>Description</u>	<u>Example</u>
3-5c(24)	SAMPEDIT	1. TEMP. UPDATE TO PERMAMENT JCL. (ALMSA.SAMJCL). 2. FOLLOWED BY DREAMST WHICH EDITS AND PUTS IT OUT ON THE SPOOL.	SAMEDIT XXXX DREAMST XXXX XXXX=JOBNAME
★ 3-5c(25)	SAMFMAT	FORMATS RMATRIX	SAMFMAT
3-5c(7)	SAMPOST	POST STATISTICS.	SAMPOST
3-5c(3)	SAMREST	RESTARTS JOB AFTER ABEND.	SAMREST JN(9999) 9999=MMATRIX JOBNUM
3-5c(1)(a)	SAMRPTA SAMRPTAH	DISPLAY RMATRIX OR HARDCOPY OF RMATRIX.	SAMRPTA OR SAMRPTAH
3-5c(1)(c)	SAMRPTI SAMRPTIH	DISPLAY JOBS NOT COMPLETE HARDCOPY JOBS INCOMPLETE.	SAMRPTI OR SAMRPTIH
3-5c(1)(a)	SAMRPTJ SAMRPTJH	DISPLAY RMATRIX BY JOB HARDCOPY RMATRIX BY JOB.	SAMRPTJ OR SAMRPTJH
3-5c(1)(d)	SAMRPTK SAMRPTKH	DISPLAY INCOMP.JOBS BY JOBNAM HARDCOPY INCOMP. BY JOBNAM.	SAMRPTK OR SAMRPTKH
3-5c(1)(e)	SAMRPTR	DISPLAY ONLY JOBS WITH ABEND OR START ONLY.	SAMRPTR
3-5c(6)	SAMRUN	START MMS AFTER TERM.	SAMRUN
3-5c(10)	SAMSCAN	STARTS SCAN TO RELEASE JOBS.	SAMSCAN
3-5c(22)	SAMSPool	LIST A JOB (80 x 80) ON THE JOBSPool PACK.	SAMSPool XXXX XXXX=JOBNAME
3-5c(15)	SAMST	DISPLAY STATUS ON HOLD C, SPool PACK, & PERM. JCL.	SAMST ENTER JOBNAMES AFTER PROMPT JOB1, JOB2, JOB3 /*
3-5c(16)	SAMSTART	ACTIVATE A JOB EVEN IF MMS IS QUIESCE.	SAMSTART XXXX XXXX=JOBNAME
3-5c(5)	SAMTERM	TERM OR QUIESCE MMS.	SAMTERM

<u>Paragraph</u>	<u>CLIST</u>	<u>Description</u>	<u>Example</u>
3-5c(11)	SAMUPD	UPDATES RUNNING MATRIX (RMATRIX).	SAMUPD.
2-4(3)	SASSFLY	ADD/DELETE JOBS ON MATRIX	SASSFLY
* 2-4(3)	SASSRUN	CREATE NEW RMATRIX	SASSRUN
* 2-4(3)	SASSSIM	SIMULATE PROCESSING OF JOBS ON RMATRIX	SASSSIM
* 2-4(2)	SASSUPD	UPDATE RMATRIX	SASSUPD
2-4c(5)(c)	SCAMINQ	DISPLAY MASTER MATRIX (MMATRIX) INFOR FOR SPECIFIC APPL. NAME(S).	SCAMINQ ENTER APPL NAME AFTER PROMPT APPL1, APPL2, APPL3 /*
2-4c(5)(c)	SCANINQ	DISPLAY MMATRIX INFO FOR SPECIFIC APPL. NUMBERS.	SCANINQ ENTER APPL. NUMBER AFTER PROMPT APPL1, APPL2, /*
2-4c(5)(c)	SCJMINQ	DISPLAY MMATRIX FOR SPECIFIC JOB NAMES	SCJMINQ ENTER JOBNAMES AFTER PROMPT JOB1, JOB2, /*
2-4c(5)(c)	SCJNINQ	DISPLAY MMATRIX INFO FOR SPECIFIC JOB NUMBERS.	SCJNINQ ENTER JOB NUMBER AFTER PROMPT JOBN01, JOBN02 /*

Appendix D

LIST OF MMS CATALOGED PROCEDURES

<u>Paragraph</u>	<u>Procedure</u>	<u>Description</u>
4-3	DREAMS	OS/HASP RDR TO SAMSPool
4-3	DREAMSE	UPDATES MASTER MATRIX RES FROM ALPHA.WJCL (XXXX)
4-3	DREAMSH	RDR TO SAMSPool
*4-3	DREAMSP	ALPHA.JCL TO ALMSA.MMS.EXECJCL UPDATES MASTER MATRIX
4-3	DREAMSR	SCRATCHES DREAMS DATA SETS AND STARTS DREAMS USES RDR ECC
3-5	SAMABNP SAMAUTOP	POST ABENDS TO A JOB AUTOMATIC RESTART OF SAMSCAN
3-5	SAMCANCP	CANCELS JOB(s) FROM JOB SPOOL
	SAMCOOPY	ADDS JOB(s) TO JOBSPOOL
3-5	SAMDISPP	DISPLAYS JOB(s) ON JOB SPOOL
3-5	SAMDOWNP	POST ABENDS TO DOWNED CPUS
3-5	SAMENDP	POST END TO R-MATRIX
3-5	SAMENVP	DISPLAYS RESOURCES AVAILABLE TO CPU
	SAMGENRP	PULLS OFF EXEC JCL TO SAMSPool
3-5	SAMPOSTP	POST STATS TO M-MATRIX
3-5	SAMRESTP	REMOVES ABEND POSTING AND SCANS
3-5	SAMRPTP	REPORTS OF ALL JOBS
3-5	SAMRUNP	REMOVES TERM POSTING AND SCANS
3-5	SAMSCANP	SCANS R-MATRIX TO RELEASE JOBS
3-5	SAMSIMP	SIMULATION OF CURRENTLY RUNNING JOBS
3-5	SAMTERMP	QUIESCES R-MATRIX

Appendix D--Continued

<u>Paragraph</u>	<u>Procedure</u>	<u>Description</u>
3-5	SAMUPDP	UPDATES R-MATRIX
2-4	SASSCHKP	SAME AS SASSIMP PLUS INCOMPLETED JOBS ON R-MATRIX
2-4	SASSFLYP	SAME AS SASSRUNP PLUS USES INCOMPLETED JOBS ON R-MATRIX
2-4	SASSFMTP	CREATES AND FORMATS M-MATRIX FOR SUBSEQUENT UPDATE PROCESSING.
2-4	SASSRPTP	PRINT SELECTED OR ALL JOBS OF MASTER DATA BASE SEGMENTS
2-4	SASSRUNP	INITIALLY START UP A SCHEDULING PERIOD
2-4	SASSSIMP	SIMULATE PROCESSING OF JOBS
2-4	SASSUPDP	UPDATE MASTER DATA BASE
2-5	SCCOMPRP	COMPARE TWO DATA BASES
2-4	SCXREFP	PRINT M-MATRIX PERTAINING TO APPLICATION NAME, NUMBER, OR JOBNAME

Appendix E

MMS LOG-ON PROCEDURE

```

000100 //MMS TSO      PROC
000200 //IEFPROC     EXEC      PGM=IKJEFT01,DYNAM=NR=25,TIME=1439,REGION=1324K
000300 //SYSPROC      DD        DSN=ALMSA.MMS.CLIST,DISP=SHR
000400 //              DL        DSN=ALMSA.USER.CLIST,DISP=SHR
000500 //              DD        DSN=ALMSA.CLIST,DISP=SHR
000600 //ISPFLLB      DD        DSN=SYS1.ISPFLLB,DISP=SHR
000700 //              DD        DSN=ALMSA.ISPFLLB,DISP=SHR
000800 //ISPFLLB      DL        DSN=SYS1.ISPFLLB,DISP=SHR
000900 //ISPFLLB      DD        DSN=SYS1.ISPFLLB,DISP=SHR
001000 //              DD        DSN=ALMSA.ISPFLLB,DISP=SHR
001100 //ISPPARM      DD        DSN=ALMSA.ISPPARM,DISP=SHR
001200 //ISPFCTL1     DD        DISP=NEW,UNIT=VIO,SPACE=(CYL,(1,1)),
001300 //              DCB=(LRECL=80,BLKSIZE=200,RECFM=FB)
001400 //ISPFCTL2     DD        DISP=NEW,UNIT=VIO,SPACE=(CYL,(1,1)),
001500 //              DCB=(LRECL=80,BLKSIZE=200,RECFM=FB)
001600 //ISPLST1      DD        DISP=NEW,UNIT=VIO,SPACE=(CYL,(1,1)),
001700 //              DCB=(LRECL=121,BLKSIZE=1210,RECFM=FB)
001800 //ISPLST2      DD        DISP=NEW,UNIT=VIO,SPACE=(CYL,(1,1)),
001900 //              DCB=(LRECL=121,BLKSIZE=1210,RECFM=FB)
002000 //SYSHELP      DD        DSN=ALMSA.HELP,DISP=SHR
002100 //              DD        DSN=SYS1.HELP,DISP=SHR
002200 //SYSPRINT     DD        TERM=TS,SYSOUT=A
002300 //SYSTERM      DD        TERM=TS,SYSOUT=A
002400 //SYSTSPRT     DD        TERM=TS,SYSOUT=A
002500 //SYSIN        DL        TERM=TS,SYSOUT=A
002600 //SYSTSIN      DD        DDNAME=IEFBRD1
002700 //IEFBRD1      DD        TERM=TS,SYSOUT=A
002800 //CARDIN       DD        TERM=TS,SYSOUT=A
002900 //INTR         DD        SYSOUT=(A,INTNL)
003000 //PT5F001      DD        TERM=TS,SYSOUT=A
003100 //PT5F001      DD        TERM=TS,SYSOUT=A
003200 //REPORTS      DD        TERM=TS,SYSOUT=A
003300 //MESSAGES     DD        TERM=TS,SYSOUT=A
003400 //SYSOUT        DD        TERM=TS,SYSOUT=A
003500 //ASCAID01     DD        TERM=TS,SYSOUT=A
003600 //PRINT        DD        TERM=TS,SYSOUT=A
003700 //EPTOUT       DD        TERM=TS,SYSOUT=A
003800 //RMATRIX      DD        DSN=ALMSA.RMATRIX,DISP=SHR
003900 //RMATRIX      DD        DSN=ALMSA.RMATRIX,DISP=SHR
004000 //SAMPPOOL     DD        DSN=ALMSA.MS.SAMPPOOL,DISP=SHR
004100 //SAMPJCL      DD        DSN=ALMSA.MS.SAMPJCL,DISP=SHR
004200 //SCRJCL       DD        DSN=ALMSA.MS.SCRJCL,DISP=SHR
004300 //JOBSPPOOL    DD        DUMMY,DCB=(BLKSIZE=80)
004400 //EXECJCL      DD        DISP=NEW,UNIT=VIO,SPACE=(CYL,(1,1)),
004500 //              DCB=(LRECL=81,BLKSIZE=810,RECFM=FB)
004600 //SCRATCH      DD        DUMMY,DCB=(BLKSIZE=80,UNITNO=1)
004700 //COPYJCL      DD        DISP=NEW,UNIT=VIO,SPACE=(CYL,(1,1)),
004800 //              DCB=(LRECL=80,BLKSIZE=800,RECFM=FB,UNITNO=1)

```

Appendix F

MMS MESSAGES

Error
number

0000 VARIABLE

Explanation: Messages prefixed by 0000 are used when requesting a terminal user to enter data to a SASS or SAM program.

System action: Processing stops until data or a /* is entered.

User response: Enter master or running matrix maintenance or inquiry request(s).

0101 SASS PROCESSING HAS TERMINATED WITH ERRORS

Explanation: One or more errors has been encountered during the processing of a SASS program.

System action: Processing is terminated.

User response: Review other generated error messages to determine error condition(s). Correct and resubmit.

0119 SASS DATABASE FORMAT IS COMPLETE

Explanation: A new master data base has been created and formatted with low-values in preparation for initial updating or conversion.

System action: Processing continues.

User response: None.

0124 SASS PROCESSING HAS COMPLETED NORMALLY

Explanation: A SASS program has completed normally.

System action: Processing is completed.

User response: None.

0124 SASS RMATRIX HAS COMPLETED

Explanation:

1. All the jobs on the running matrix have completed normally. There is nothing left for SAM to release.

2. A simulation of incomplete jobs remaining on the running matrix has been requested, but there are no incomplete jobs on the running matrix.

Appendix F--Continued

System Action:

1. Processing is completed.
2. Processing is terminated.

User Response:

1. Execute SASSFLY(P), or SAMPOST(P) and SASSRUN(P), to establish the next schedule.
2. None.

Ø134 SASS PROCESSING WILL CONTINUE

Explanation: An error or warning condition has occurred that is not critical enough to cause immediate program termination.

System Action: Processing continues.

User Response: Corrective action may or may not be required.

Ø147 SASS RMATRIX/CMATRIX IS CREATED

Explanation:

1. The running matrix(RMATRIX) has been created as a result of successful execution of SASSRUNP.
2. A temporary running matrix (CMATRIX) is created as a result of successful execution of SASSCHKP.

System Action: Processing continues.

User Response: None.

Ø148 SASS MMATRIX IS UPDATED

Explanation: The master matrix has been updated as a result of a PMOD change (SASSUPDP) or statistics posting of jobs completed on the running matrix (SASSFLYP) or SAMPOST(P)).

System Action: Processing continues.

User Response: None.

Ø148 SASS RMATRIX IS UPDATED

Explanation:

1. The running matrix has been updated as a result of a TMOD update, fly add, fly delete, or statistics posting and purge through SASSFLYP.

Appendix F--Continued

2. The running matrix has been updated as a result of job start, end, ABEND posting by the SAMEND program, or an update by SAMUPD(P), or statistics posting and purge by SASSFLYP or SAMPOST(P).

System Action: Processing continues.

User Response: None.

Ø150 SASS RMATRIX CURRENTLY BEING QUIESCED

Explanation: The running matrix has been quiesced (termed) in order to prevent further releasing of jobs by the SAMEND program. This is the first of two messages (ref 3135).

System Action: Processing continues.

User Response: To remove the TERM execute SAMRUN(P), otherwise no response is not necessary.

Ø163 SASS RMATRIX WILL BE ACTIVATED/QUIESCED

Explanation:

1. The term is removed from the running matrix and a scan is initiated to allow jobs to release.

2. The running matrix is being quiesced (termed) to prevent the release of jobs.

System Action: Processing continues.

User Response: None.

Ø204 INPUT DATA NOT RECEIVED

Explanation: No update requests were input to the SAMUPD program.

System Action: Processing is terminated.

User Response: Submit update requests and rerun.

Ø205 INPUT FOR '___' NOT FOUND

Explanation: A control statement (e.g., ./ FLY ADD) has been processed with no input data statements.

System Action: The control statement is rejected. Processing continues.

User Response: Re-enter control statement (card) with data statements (cards) following it.

Appendix F--Continued

0223 INPUT DATA IS MISSING FOR '___'.

Explanation: An input control statement associated with the function being executed (e.g., ./ PMOD MATRIX for the execution of SASSUPDP) was not processed.

System Action: Processing continues.

User Response: Enter the control statements (cards) associated with the function being executed.

0306 DATA CARD WITH NO CONTROL CARD

Explanation: A data statement (card) has been processed with no preceding control statement (card) in the execution of a function requiring both.

System Action: The data statement (card) is rejected. Processing continues.

User Response: Correct and re-enter.

0309 DATA FOR KEYWORD '___' EXCEEDS MAX LENGTH

Explanation: The input data for a keyword exceeds the maximum length allowed.

System Action: The data statement (card) is rejected. Processing continues.

User Response: Correct and re-enter.

0310 DATA FOR KEYWORD '___' EXCEEDS COL 71

Explanation: The data for a keyword exceeds column (or position) 71.

System Action: The statement (card) is rejected. Processing continues.

User Response: Correct and re-enter.

0315 DATA CARD NOT VALID W/PRIOR ERRORS ENCOUNTERED

Explanation: An error has been detected in a prior card or continuation card that nullifies all continuation cards associated with the action to be performed.

System Action: The continuation cards are rejected. Processing continues.

User Response: Correct and re-enter the transaction and all continuation cards (statements).

Appendix F--Continued

Ø318 DATA FOR KEYWORD '___' IS INVALID_

Explanation: The data for a keyword contains an incorrect value or format.

System Action: The statement (card) is rejected. Processing continues.

User Response: Correct and re-enter.

Ø321 DATA FOR KEYWORD '___' NOT SPECIFIED

Explanation: No value was entered for a keyword.

System Action: The statement (card) is rejected. Processing continues.

User Response: Correct and re-enter.

Ø322 DATA CARD WITH NO DATA

Explanation: A statement (card) was encountered with no data prior to position (column) 72.

System Action: The statement (card) is rejected. Processing continues.

User Response: Correct and re-enter.

Ø '8 DATA FOR KEYWORD '___' REQUIRES PARENS

Explanation: Keywords that have a variable number of entries associated with them (e.g., FN=(SØ3, U52)) must have their values enclosed in parentheses.

System Action: If the values are not enclosed in parenthesis, then the statement (card) is rejected. Processing continues.

User Response: Correct and re-enter.

Ø4Ø9 PARM DATA/ENTRY NAME EXCEEDS MAX LENGTH.

Explanation: Either the entire parameter field or one of the sub-parameters exceeds the maximum length allowed.

System action: Processing is terminated.

User response: Correct parameter field entry and rerun.

Appendix F--Continued

0417 PARM DATA/ENTRY NAME IS OMITTED

Explanation: Either the entire parameter field or one of the sub-parameters has been omitted.

System action: Processing is terminated.

User response: Enter the parameter field(s) missing and rerun.

0418 PARM entry name '___' IS INVALID

Explanation: A sub-parameter is invalid.

System action: Processing is terminated.

User response: Refer to documentation in chapter 2 or 3 for correct symbolic, or parameter, values. Correct and re-enter.

0428 PARM JOBNUM '___' DOES NOT EXIST IN RMTIX

Explanation: Using the master matrix job number (JOBNUM) a scan was initiated on the running matrix for a job containing that master matrix number. No job was found.

System action: Processing is terminated.

User response: Add the job to the running matrix or correct the symbolic value and re-enter.

0504 EXPECTED CONTINUATION NOT RECEIVED

Explanation: The last statement (card) processed ended with a comma followed by a blank indicating a continuation statement (card) would follow. A continuation statement (blank in position or column 1) was not encountered.

System action: All preceding statements associated with the same request are nullified. Processing continues.

User response: Correct and re-enter.

0607 SKIPPING UNTIL NEXT CONTROL CARD

Explanation: An error has been encountered in a control statement (card).

System action: All data statements (cards) following the control statement in error are bypassed.

User response: Correct and re-enter the control and data statements (cards).

Appendix F--Continued

Ø702 OPERATION '___' NOT PERMITTED WITH PARM REQUEST

Explanation: The control statement (card) operation specified is mutually exclusive with the execution function.

System action: The control statement (card) is rejected. Processing continues.

User response: Refer to Table 2-1 for allowable operations. Correct and re-enter.

Ø709 OPERATION STARTING IN COL '___' EXCEEDS MAX LENGTH

Explanation: The control statement (card) operation exceeds the maximum length allowed.

System action: The control statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

Ø710 OPERATION EXCEEDS COL 71

Explanation: The control statement (card) operation exceeds position (column) 71.

System action: The control statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

Ø718 OPERATION '___' IS INVALID

Explanation: The operation specified on the control statement (card) is invalid.

System action: The control statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

Ø721 OPERATION NOT SPECIFIED

Explanation: No operation was specified on a control statement (card).

System action: The control statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

Appendix F--Continued

0802 OPERAND NOT PERMITTED WITH RUN

Explanation: No operand is permitted if the control statement (card) operation is RUN.

System action: The control statement (card) is rejected. Processing continues.

User response: Remove the operand and re-enter.

0809 OPERAND STARTING IN COL '___' EXCEEDS MAX LENGTH

Explanation: The control statement (card) operand exceeds the maximum length allowed.

System Action: The control statement (card) is rejected. Processing continues.

User Response: Correct and re-enter.

0810 OPERAND EXCEEDS COL 71

Explanation: The control statement (card) operand exceeds position (column) 71.

System Action: The control statement (card) is rejected. Processing continues.

User Response: Correct and re-enter.

0818 OPERAND '___' IS INVALID

Explanation: The control statement (card) is invalid for the operation specified.

System Action: The control statement (card) is rejected. Processing continues.

User Response: Correct and re-enter.

0821 OPERAND NOT SPECIFIED

Explanation: No operand was specified on a control statement (card) requiring one.

System Action: The control statement (card) is rejected. Processing continues.

User Response: Refer to chapter 2 for valid operands for each operation. Correct and re-enter.

Appendix F—Continued

0914 PMOD IS MUTUALLY EXCLUSIVE WITH PRIOR TMOD

Explanation: The operation TMOD has already been encountered in the input stream. Temporary and permanent master matrix updates may not be mixed in one execution.

System Action: The control statement (card) is rejected. Processing continues.

User Response: Submit the permanent changes in a separate execution of SASSUPDP.

1014 TMOD MUTUALLY EXCLUSIVE WITH PRIOR PMOD

Explanation: The operation PMOD has already been encountered in the input stream. Permanent and temporary master matrix updates may not be mixed in one execution.

System Action: The control statement (card) is rejected. Processing continues.

User Response: Submit the temporary changes in a separate execution of SASSUPDP.

1102 KEYWORD OTHER THAN AC, JN, AND JM/OTHER THAN AC, AN, AND AM/OTHER THAN AC, FN, AND FM NOT PERMITTED WITH DELETE

Explanation: When deleting a master matrix job, application, or file; only the action of delete and the job number and name, application number and name, or file number and name may be specified.

System action: The data statement (card) is rejected. Processing continues.

User Response: Correct and re-enter.

1102 KEYWORD OTHER THAN JN NOT PERMITTED WITH RM

Explanation: When requesting that a running matrix job be deleted (RM=DELETE) or cleared (RM=CLEAR) through SAMUPD(P), only the keywords RM and JN are allowed.

System Action: Data statement (card) is rejected. Processing continues.

User Response: Correct and re-enter.

Appendix F--Continued

1102 KEYWORD RM NOT PERMITTED WITH ADD

Explanation: When the SAMUPD program has determined that the update request is a job add, then the keyword RM is not permitted.

System Action: Data statement (card) is rejected. Processing continues.

User Response: Correct and re-enter.

1109 KEYWORD STARTING IN COL'__' EXCEEDS MAX LENGTH

Explanation: The keyword exceeds the maximum length allowed.

System action: The data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

1110 KEYWORD STARTING IN COL'__' EXCEEDS COL 71

Explanation: The keyword exceeds position (column) 71.

System action: The data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

1113 KEYWORD '___' MUTUALLY EXCLUSIVE WITH

Explanation:

1. Keyword "FS" (files reset) is mutually exclusive with all other keywords.

2. Keyword "ND" (set the number of devices for a file) is mutually exclusive with all other keywords.

System action: Data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

1118 KEYWORD STARTING IN COL '___' IS INVALID

Explanation: An unrecognizable keyword has been encountered on a data statement (card).

System action: The data statement (card) is rejected. Processing continues.

User Response: Correct and re-enter.

Appendix F--Continued

1121 KEYWORD '___' NOT SPECIFIED

Explanation: A keyword required for a particular action to be performed was not specified.

System action: In data statement (card) is rejected. Processing continues.

User response: Supply required keyword (ref chap 2 or 3) and re-enter.

1131 KEYWORD '___' IS DUPLICATE

Explanation: A duplicate keyword has been encountered on the same data statement (card) or continuation.

System action: The data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

1158 KEYWORD '___' MUST BE SPECIFIED FOR ADD

Explanation: The keywords JN(job number), DN (master matrix job number), and JM (job name) must be specified for a job add to the running matrix through the SAMUPD program. One of these keywords is missing.

System action: Data statement (card) is rejected.

User response: Correct and re-enter.

1279 TRANS BYPASSED

Explanation: A data statement following a rejected control statement is bypassed in processing.

System action: Data statements are bypassed until a control statement or end of file is encountered.

User response: Correct the control statement (card), and re-enter both the control and associated data statement(s).

1326 VALUE FOR KEYWORD '___' LESS THAN MINIMUM

Explanation: The numeric value for a keyword is less than the minimum allowed.

System action: The data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

Appendix F--Continued

1327 VALUE FOR KEYWORD '___' GREATER THAN MAXIMUM

Explanation: The numeric value for a keyword is greater than the maximum allowed.

System action: The data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

1416 DEPENDENCY ERROR(S) ENCOUNTERED DURING SIMULATION

Explanation: One or more dependency errors have been encountered during simulation (e.g., job backs off to itself). This is the first of three messages (ref 3100 (INITS EMPTY) and 3136).

System action: Processing is terminated.

User response: Review error messages; correct job(s) dependency field(s) in the master matrix and rerun.

1418 DEPENDENCY NUMBER '___' IS INVALID

Explanation:

1. The dependency number specified on the data statement to update the master matrix does not fall in the range of absolute numbers 1 through 3525.

2. The dependency number specified on the data statement to update the running matrix does not fall in the range of 1 through 555.

System action: The data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

1428 DEPENDENCY '___' DOES NOT EXIST IN MMATRIX/RMATRIX

Explanation:

1. In a master matrix update, the job specified by the dependency number does not exist (is not in use) in the master matrix.

2. In a running matrix update, the job specified by the dependency number does not exist (is not in use) in the running matrix.

System action: The data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

Appendix F--Continued

1431 DEPENDENCY NUMBER '___' IS DUPLICATE IN PARAMETER LIST

Explanation: A dependency number in the keyword DP parameter list occurs more than once.

System action: The data statement (card) is rejected. Processing continues.

User response: Remove duplicate dependency number(s) and re-enter.

1480 DEPENDENCY NUMBER '___' MAY NOT EQUAL JOBNUM

Explanation: A job number to be updated may not specify a dependency equal to itself.

System action: The data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

1518 NUMBER DEVICES '___' IS INVALID

Explanation: In the running matrix file update, the number of devices specified for the ND keyword is not numeric (from 1 through 99).

System action: Data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

1527 NUMBER OF DEPENDENCIES/OF FILES GR THAN MAXIMUM

Explanation:

1. The number of dependencies specified for a job exceeds the maximum of 12.

2. The number of files specified for a job exceeds the maximum of 9.

System action: The data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

Appendix F--Continued

1541 NUMBER OF DEPS FOR JOB '___' EXCEEDS MAXIMUM

Explanation: The dependency optimizer process (executes before the actual job simulation), in determining a job's dependencies from the jobs that are scheduled, found more than 12 dependencies.

System action: Processing is terminated.

User response: TMOD the 12 most important dependencies to the job and re-run.

1602 FILE REPORT NOT PERMITTED WITH TERMINAL MODE

Explanation: A control statement requesting the files report (./REPORT FILES) was encountered while executing through TSO. This option is not permitted under TSO because of the report width.

System action: Control statement is bypassed. Processing continues.

User response: Create the report through batch.

1618 FILE NUMBER/MODE/STATUS '___' IS INVALID

Explanation:

1. The file number specified on the data statement (card) does not fall in the range of numbers 1 through 255.

2. The file mode specified on a job update data statement (card) does not equal R (reference), U (update), S (share), or E (exclusive-use).

3. The file status specified on a file segment update against the running matrix does not equal L (lock), X (lock except for reference users), or F (clear the current file status and reset with current status, and number of users, based on a scan of the running matrix).

System action: Data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

1628 FILE NUMBER '___' DOES NOT EXIST IN MMATRIX

Explanation: The file number specified is not active in the master matrix.

Appendix F--Continued

System action: The data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

1650 FILE '___' CURRENTLY BEING UPDATED/SHARED/LOCKED/XLOCKED/EXCLUDED/REFERENCED

Explanation: The user has requested that a start be posted to a job on the running matrix. The job requests a file in a mode that is in conflict with the current file status. This is the first of two messages (ref 3118).

System action: The data statement (card) is rejected. Processing continues.

User response: Change the status of the file by posting other jobs as complete, remove file lock or XLOCK, etc., and re-enter.

1653 FILE NAME/NUMBER '___' ALREADY EXISTS IN MMATRIX

Explanation

1. The file name already exists in the master matrix invalidating the file add or name change request.

2. The file number specified on a file add is already in use in the master matrix.

System action: The data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

1654 FILE NUMBER '___' IS DUPLICATE IN PARAMETER LIST

Explanation: A file number in the keyword 'FN' parameter list occurs more than once.

User response: Remove duplicate file number(s) and re-enter.

1666 FILE NAME '___' DOES NOT MATCH MMATRIX

Explanation: The file name specified on a file delete data statement does not match the master matrix file name.

System action: The data statement is rejected. Processing continues.

User response: Correct and re-enter.

Appendix F--Continued

1733 EQUAL SIGN IN COL '___' WITH NO PRECEDING KEYWORD

Explanation: An equal sign was encountered on a data statement with no preceding keyword.

System action: The data statement is rejected. Processing continues.

User Response: Correct and re-enter.

1802 APPL REPORT NOT PERMITTED WITH TERMINAL MODE

Explanation: A control statement requesting the application report (./ REPORT APPL) was encountered while executing through TSO. This option is not permitted under TSO because of the report width.

System action The control statement is bypassed. Processing continues.

User response: Create the report through batch.

1828 APPL NAME/NUMBER '___' DOES NOT EXIST IN MMATRIX

Explanation:

1. The application name specified does not exist in the master matrix.

2. The application number specified does not exist in the master matrix.

System action: The data statement (card) is rejected. Processing continues

User response: Correct and re-enter.

1853 APPL NAME/NUMBER '___' ALREADY EXISTS IN MMATRIX

Explanation:

1. The application name specified on an application add or name change already exists in the master matrix.

2. The application number specified on an application add is already in use in the master matrix.

System action: The data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

Appendix F--Continued

1866 APPL NAME '___' DOES NOT MATCH MMATRIX

Explanation: The application name specified on an application delete data statement does not match the master matrix application name.

System action: The data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

1900 JOB ADDED/DELETED/CLEARED/UPDATED

Explanation: This message is displayed after each user request against a job on the running matrix has been accomplished (through SAMUPD(P)).

System action: Processing continues.

User response: None.

1921 JOB CHANGE NOT SPECIFIED

Explanation: No keyword other than JM (job number) was entered on the data statement (card).

System action: Data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

1924 JOB '___' HAS COMPLETED NORMALLY/ABNORMALLY/PREVIOUSLY

Explanation:

1. A job has completed normally or abnormally.
2. The running matrix job entry reflects that the job has already been posted with a job completion.

System action:

1. The job entry is posted with completion type, time, and elapsed time. If the job ABENDs field is updated and files used in update, share, or exclusive-use mode are locked, a scan is initiated against the running matrix for jobs to release.
2. No posting is done to the job entry. A scan is initiated against the running matrix for jobs to release.

User response: None.

Appendix F--Continued

1928 JOB NAME/NUMBER '___' DOES NOT EXIST IN MMATRIX

Explanation:

1. The job name to be fly added, fly deleted, or run does not exist in the master matrix.
2. The job number specified on the data statement (card) does not exist in the master matrix.

System action:

1. Processing continues, but if processing in batch mode, the fly (check) or run and the simulation are terminated.
2. The data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

1928 JOB '___' DOES NOT EXIST IN MMATRIX

Explanation:

1. A completed job on the running matrix does not exist on the master matrix (may have been a one time only add).
2. An incomplete job on the running matrix does not exist on the master matrix.

System action:

1. The job's run statistics are not posted to the master matrix. Processing continues.
2. The job is included in the simulation, and processing continues

User response: None.

1928 JOB '___' DOES NOT EXIST IN RMATRIX

Explanation: The job number specified on an update request against the running matrix does not exist (is not in use).

System action: Data statement (card) is rejected. Processing continues.

User response: If the job is to be added, specify JM (jobname) and DN (master matrix job number); otherwise correct the JN (job number), and re-enter.

Appendix F—Continued

1928 JOB TO BE FLY DELETED '___' DOES NOT EXIST IN RMATRIX

Explanation: A job name to be fly deleted is not found on the running matrix.

System Action: Processing continues.

User response: None.

1928 JOB NAME '___' DOES NOT EXIST IN RMATRIX/HOLD QUEUE

Explanation:

1. The job name to be processed by the SAMEND program does not exist in the running matrix.

2. A job on the running matrix to be released under HASP does not exist in the hold queue.

System action: Processing is terminated.

User response:

1. Add the job to the running matrix and rerun.

2. Read the job into the hold queue and rerun.

1937 JOB NAME(S) TO BE RUN NOT SPECIFIED

Explanation: No job or application names were specified on the data statement (card) following the ./ RUN control statement.

System action: Processing continues, but the run and simulation functions are terminated.

User response: Specify job or application name(s) and re-enter.

1938 JOB NAME(S) TO BE FLY ADDED NOT SPECIFIED

Explanation: No job or application names were specified on the data statement (card) following the ./ FLY ADD control statement.

System action: Processing continues, but the fly (check) and simulation functions are terminated.

User response: Specify job or application names and re-enter.

Appendix F--Continued

1939 JOB NAME(S) TO BE FLY DELETED NOT SPECIFIED

Explanation: No job or application names were specified on the data statement (card) following the ./ FLY DELETE control statement.

System action: Processing continues, but the fly (check) and simulation functions are terminated.

User response: Specify job or application names and re-enter.

1939 JOB '___' TO BE FLY DELETED DOES NOT EXIST IN RMATRIX

Explanation: A job specified on the data statement (card) to be fly deleted does not exist on the running matrix.

System action: Processing continues.

User response: None.

1939 JOB '___' TO BE FLY DELETED IS EXECUTING

Explanation: A job specified on the data statement (card) to be fly deleted is represented on the running matrix as still executing (i.e., start and no end or ABEND is posted). The job is not deleted from the running matrix. This is the first of two messages (ref. 3159, DELETE NOT INVOKED).

System action: Processing continues.

User response: Either delete the job through SAMUPD(P), or post a completion to the job through SAMEND(P). In the latter option, SASS will automatically delete the job from the running matrix in the next execution of SASSFLY(P).

1941 JOB '___' EXCEEDS MAXIMUM SYSTEM RESOURCES

Explanation: A job to be simulated requests more than the system maximum amount (number) of a particular resource (e.g., Tape 9, CORE).

System action: Processing is terminated.

User response: Review the resources required by the job in the master matrix, or run matrix and correct the resource requirement that is in error.

1950 JOB '___' CURRENTLY BEING RESTARTED

Explanation: An ABENDED job is being restarted through the execution of SAMREST(P).

Appendix F--Continued

System action: The job is restarted. File locks are removed from files that the job used in share, update, or exclusive-use mode. A scan is initiated against the running matrix for other jobs to release.

User response: None.

1953 JOB NAME/NUMBER ALREADY EXISTS IN MMATRIX

Explanation:

1. The job name specified on a job add or name change data statement already exists in the master matrix.
2. The job number specified on a job add is already in use in the master matrix.

System action: The data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

1953 JOB NAME '___' ALREADY EXISTS IN RMATRIX

Explanation: The job name specified on a job add or name change already exists in the running matrix for a different job number.

System action: Data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

1953 JOB TO BE FLY ADDED '___' ALREADY EXISTS IN RMATRIX.

Explanation: The job name, specified on a data statement, to be fly added already exists on the running matrix.

System action: Processing continues.

User response: None.

1964 JOB '___' HAS ABENDED PRIOR TO NORMAL END

Explanation: A job that has ABENDED prior to normal end is automatically deleted from the running matrix in the execution of the fly function, but no statistics are posted to the master matrix. This is the first of three messages (ref 3167 and 3170).

System action: Processing continues.

User response: None.

Appendix F-Continued

1966 JOB NAME '___' DOES NOT MATCH MMATRIX

Explanation: The job name specified on a job delete data statement does not match the master matrix job name.

System action: The data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

1966 JOB '___' DOES NOT MATCH MMATRIX

Explanation:

1. A completed job on the running matrix does not match the master matrix jobname. Statistics are not posted to the master matrix, but the job is deleted from the running matrix. This is the first of three messages (ref 3167 and 3170).

2. An incomplete job on the running matrix does not match the master matrix job name.

System action:

1. Processing continues.
2. Processing is terminated.

User response:

1. None.
2. Change the running matrix job's master matrix number (through SAMUPD(P)) to a master matrix number not in use, and re-run.

1967 JOB START NOT POSTED TO RMATRIX

Explanation: User is trying to post an end or ABEND to a job on the running matrix which has not been posted with a start. This is the first of two messages (ref 3118).

System action: Data statement (card) is rejected. Processing continues.

User response: Add a start time to the update request and re-enter.

Appendix F—Continued

1968 JOB '___' PROCESSED TO EOJ

Explanation: All job names printed from the running matrix have processed to normal completion under the control of SCRAM. This is the first of two messages (ref 3182). Run statistics are posted to the master matrix.

System action: Processing continues.

User response: None.

1968 JOB '___' PROCESSED TO EOJ OUTSIDE SASS

Explanation: All job names printed from the running matrix have processed to normal completion, but was not started under SCRAM control. Run statistics are not posted to the master matrix. This is the first of two messages (ref 3167).

System action: Processing continues.

User response: None.

1970 JOB '___' DELETED FROM RMATRIX

Explanation: In the execution of the fly (or chk) process this message is produced whenever SASS deletes a job from the RMATRIX, either automatically because the job is completed or by user request.

System action: Processing continues.

User response: None.

1973 JOB '___' SELECTED - NOT RELEASE AT '___'

Explanation: A job has ben selected as available for release in simulation, or actual system scheduling, but external criteria prevent its release (e.g., initiators unavailable, core, devices unavailable, file(s) used in conflicting mode, etc.) The time that the job was selected is displayed. This is the first of three messages (ref number 3149, 3150, 3171, and 3172).

System action: Processing continues.

User response: None.

1976 JOB '___' IS EXECUTING

Appendix F--Continued

Explanation:

1. A TMOD was attempted during a fly (or chk) process against a job that is executing. This is the first of two messages (ref 3159, TMOD not invoked).
2. An update, other than posting an end or ABEND, was attempted against an executing job through SAMUPD(P). This is the first of two messages (ref 3118).

System action:

1. The TMOD is not invoked. Processing continues.
2. The data statement (card) is rejected. Processing continues.

User response:

1. None.
2. Post an end, or clear the job, before re-entering the update.

1982 JOB START HAS OCCURRED.

Explanation: An attempt was made to post an early start time to a job with a start time already posted. This is the first of two messages (ref 3118).

System action: Data statement (card) is rejected. Processing continues.

User response: Clear the job (RM=CLEAR) before posting an early start.

1983 JOB '___' BACKS OFF TO JOB '___'

Explanation: A dependency error has been encountered in the simulation process. This is the first of three messages (ref 3183 and 3184) showing the back-off of dependencies until a job is found depending on the first job referenced in this message, which would create a loop if not intercepted.

System action: Processing is terminated.

User response: Correct one or more jobs' dependency fields and re-run.

1993 JOB '___' WAS RELEASED

Explanation: A job has been released for execution by the SAMEND program.

Appendix F--Continued

System action: Processing continues.

User response: None.

★ 1993 JOB '___' WAS RELEASED OUTSIDE MMS

Explanation: A job entry on the running matrix, to be updated with completion data, did not have a start posted to it.

System action: The end or ABEND, and stop time are posted. No elapsed time is computed. A scan is initiated against the running matrix for jobs to release.

User response: None.

1994 JOB '___' CURRENT SHARE USER ON FILE '___'

★ Explanation: When a job under MMS control ABENDs and is using file(s) in share mode, all other active users of the file(s) in share mode are displayed to the user.

System action: Processing continues.

User response: None.

1998 JOB '___' CAN BE RELEASED

Explanation: During the execution of the SAMSCAN function against a "termed" running matrix, a job is found as available for release.

System action: The job is not released. Processing continues.

User response: None.

2037 MAX NUMBER OF JOBS TO BE RUN EXCEEDS 555

Explanation: An attempt has been made to fly (chk) add jobs to the running matrix, bringing the total number of jobs over 555.

System action: Processing continues, but if executing in batch mode, the fly (or chk) and simulation functions are terminated.

User response: Correct and re-enter.

2039 MAX NUMBER OF JOBS TO BE FLY DELETED EXCEEDS 555

Explanation: An attempt has been made to fly delete more than 555 jobs. This exceeds the number of jobs possible on the running matrix.

Appendix F--Continued

System action: Processing continues, but if executing in batch mode, the fly and simulation functions are terminated.

User response: Correct and re-enter.

2101 SAM PROCESSING HAS TERMINATED WITH ERRORS

* Explanation: A program in the SAM portion of MMS has terminated because of error(s) already reported to the user.

System action: Processing is terminated.

User response: Review error messages, correct, and rerun.

2124 SAM PROCESSING HAS COMPLETED NORMALLY

* Explanation: A program in the SAM portion of MMS has completed normally.

System action: Processing is completed.

User response: None.

2721 CHANGE DATA NOT SPECIFIED

Explanation: A required keyword was specified on a data statement, but no optional keywords followed.

System action: The data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

2858 ACTION OF CHANGE MUST BE SPECIFIED

Explanation: An action other than CHANGE was specified on a data statement (card) for a DEVICES or ENV update.

System action: The data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

2959 SIMULATOR NOT INVOKED

Explanation: One or more errors have been encountered in the execution of SASSIMP, SASSRUNP, SASSFLYP, SASSCHKP, or SAMSIMP that were critical enough to terminate or bypass simulation.

System action: Processing is terminated.

User response: Review and correct errors encountered by SASS rerun.

Appendix F--Continued

3100 XXXX XXXX XXXX XXXX XXXX

Explanation: This message is used by the SAMEND program to display blocks of core that are contiguous in the system. It is message two and three of three messages (ref 3586).

System action: Processing continues.

User response: None.

3100 INITS EMPTY

Explanation: Dependency errors have occurred in the simulation processing preventing the release of the unsimulated jobs. This is the second of three messages (ref 1416 and 3136).

System action: Processing is terminated.

User response: Review error messages. Correct the jobs dependency fields in the master matrix and rerun.

3105 INITIATORS NOT FOUND AVAILABLE

Explanation: No class A initiators were found that were available for releasing a job.

System action: Processing continues.

User response: Add class A to more initiators; or no action is required.

3118 REQUEST IS INVALID

Explanation: An update request through SAMUPD(P), though syntactically correct, is invalid because of conflicting conditions on the running matrix. This is the second of two messages (ref 1976, 1650, 1967, and 1982).

System action: Data statement (card) is rejected. Processing continues.

User response: Take action as appropriate.

3135 SCAN PROCESSING BYPASSED

Explanation: The scan of the running matrix, for jobs to release, is bypassed because the running matrix has been quiesced (termed).

System action: Processing continues.

User response: None.

Appendix F--Continued

3136 TOTAL NUMBER OF JOBS NOT RUN

Explanation: Dependency errors have occurred in the simulation processing. This is the third of three messages (ref 1416 and 3100 (INITS EMPTY)).

System action: Processing is terminated.

User response: Review error messages; correct dependency errors; and rerun.

3141 JOB '___' EXCEEDS MAXIMUM

Explanation: After the maximum number of jobs to be run, fly added, or fly deleted has been reached, any additional jobs specified will be displayed.

System action: Processing continues, but if executing in batch mode, the run or fly and simulation functions are terminated.

User response: Correct and re-enter.

3149 FILE '___' REQUESTED AS SHARE/UPDATE/EXCL-USE/REFERENCE

Explanation: A job has been selected as available for release in the simulation or actual system scheduling process, but a file requested by the job is already in use in an incompatible mode. Incompatible requests and current use are:

<u>Incompatible requests</u>	<u>Current use</u>
share	update exclusive-use
update	share update exclusive-use
exclusive-use	reference update share exclusive-use
reference	exclusive-use

This is the second of three messages (ref 1973 and 3150).

System action: Processing continues.

User response: None.

Appendix F--Continued

3150 FILE '___' CURRENTLY BEING SHARED/UPDATED/REFERENCED/EXCL-USED

Explanation: A job has been selected as available for release in the simulation, or actual system scheduling process, but a file requested by the job is already in use in an incompatible mode. This is the third of three messages (ref 1973 and 3149).

System action: Processing continues.

User response: None.

3159 DELETE NOT INVOKED

Explanation: A job specified on the data statement (card) to be fly deleted is represented on the running matrix as still executing. The job is not deleted from the running matrix. This is the second of two messages (ref 1939).

System action: Processing continues.

User response: Either delete the job through SAMUPD(P), or post a completion to the job through SAMEND(P) or SAMUPD(P). In the latter option, SASS will automatically delete the job from the running matrix in the next execution of SASSFLYP.

3163 POSTING WILL BE BYPASSED

Explanation: The job entry in the running matrix has already been posted as complete or ABENDED.

System action: No posting to the run matrix is done. A scan against the running matrix is initiated for jobs to release.

User response: It may be necessary to remove file locks, or post normal completion through SAMUPD(P), if the job had previously ABENDED.

3167 NOT POSTED TO MMATRIX

*

Explanation: In the execution of the fly (chk) function, an ended or ABENDED job on the running matrix was not posted to the master matrix because the job did not exist on the master matrix, the job ABENDED prior to a normal end, the running matrix jobname did not equal the master matrix jobname, or the job was started outside of MMS control. This is the second of two messages (ref 1928, 1964, 1966, or 1968 (Job processed to EOJ outside SASS)).

System action: Processing continues.

User response: None

Appendix F--Continued

3170 DELETED FROM RMATRIX

Explanation: In the execution of the fly (or chk) function, a normally completed job will be deleted from the running matrix automatically.

System action: Processing continues.

User response: None.

3171 INITIS/CORE/DISK/TAPE9/TAPE7/PRINT/PUNCH REQUESTED ' _ '

Explanation: A job has been selected as available for release in the simulation, or actual system scheduling process, but the core, initiator, or devices required by the job are not available. The amount or number requested is displayed. This is the second of three messages (ref 1973 and 3172).

System action: Processing continues.

User response: None.

3172 INITIS/CORE/DISK/TAPE9/TAPE7/PRINT/PUNCH AVAILABLE ' _ '

Explanation: A job has been selected as available for release in the simulation or actual system scheduling process, but the core, initiator, or devices required by the job are not available. The amount or number available is displayed. This is the third of three messages (ref 1973 and 3171).

System action: Processing continues.

User response: None.

3179 PARAMETER BYPASSED

★ Explanation: A field for inquiry on the data statement input to the MMS inquiry program was in error. The field is bypassed.

System action: Processing continues.

User response: Correct and re-enter.

3182 POSTING HAS OCCURRED

★ Explanation: A normally completed job, executed under MMS control, is posted to the master matrix. This is the second of two messages (ref 1968 (job processed to EOJ)).

System action: Processing continues.

User response: None.

Appendix F--Continued

3183 WHICH BACKS OFF TO JOB '___'

Explanation: A dependency error has been encountered in the simulation process. This is the second of three messages (ref 1983 and 3184). This message will be displayed for every job in the back-off process until the jobs causing the dependency loop are found.

System action: Processing is terminated.

User response: Correct one or more jobs' dependency fields and rerun.

3187 TAPE9(S)/TAPE7(S)/DISK(S)/PRINTER(S)/PUNCH(S) = '___'

Explanation: In the execution of the SAMEND program the total number of each device type is displayed. This is the second through sixth message (ref 3600).

System action: Processing continues.

User response: None.

3187 REJECTS = '___'

Explanation: When the program SASS and SAMUPD are executing in batch mode, the total number of input records rejected is displayed at the end of input processing.

System action: Processing continues.

User response: None.

3189 INACTIVE = '___'

Explanation: The total number of inactive initiators is displayed by execution of the SAMEND program.

System action: Processing continues.

User response: None.

3190 INACTIVE CLASS A = '___'

Explanation: The total number of inactive initiators, with an assigned class A, is displayed by the execution of the SAMEND program.

System action: Processing continues.

User response: None.

Appendix F--Continued

3191 INACTIVE CLASS '___'

Explanation: Each inactive initiator, with its assigned classes, is displayed by the execution of the SAMEND program.

System action: Processing continues.

User response: None.

3195 TOTAL = '___'

Explanation: The total number of initiators is displayed by the execution of the SAMEND program.

System action: Processing continues.

User response: None.

3277 OPTIMUM SYSTEM UTILIZATION NOT EFFECTED AT '___'

Explanation: In the simulation or actual system scheduling process, more jobs are available for release than there are initiators available. The time is displayed.

System action: Processing continues.

User response:

1. Increase the number of initiators for the simulation (maximum of 10).
2. Assign a class A to more system initiators.

3382 OVERFLOW IN TABLE '___' HAS OCCURRED

Explanation: One of two fixed tables has been exceeded in the dependency optimizer process.

System action: Processing is terminated.

User response: Contact USADARCOM ALMSA (DRXAL-TA) immediately.

3382 OVERFLOW IN TABLE '___' HAS OCCURRED FOR JOB '___'

Explanation: The master matrix cross-reference program has exceeded the limit of its fixed table of extracted dependent jobs for a job being cross-referenced.

System action: The jobs extracted, up to the limit, for that job will be reported. Processing continues.

User response: Contact USADARCOM ALMSA (DRXAL-TA) immediately.

Appendix F--Continued

3475 AVAILABLE SYSTEM RESOURCES AT '___'

Explanation: When the SAMEND program is executed, the available system resources are displayed. This message, the first of several, displays the time of execution.

System action: Processing continues.

User response: None.

3586 CONTIGUOUS BLOCKS OF CORE

Explanation: This is the header message used by the SAMEND program to display blocks of core that are contiguous in the system. It is the first of three messages (ref 3100).

System action: Processing continues.

User response: None.

3600 DEVICES

Explanation: This is the header message used by the SAMEND program to display devices available in the system. It is the first of six messages (ref 3187).

System action: Processing continues.

User response: None.

3700 INITIATORS

Explanation: This is the header message used by the SAMEND program to display total, active, inactive, and inactive by class initiators in the system (ref 3195, 3189, 3190, and 3191).

4549 THERE IS NO CURRENT SHARE USER IN FILE '___'

Explanation: When a job ABENDs, and is using a file in the share mode, a scan is initiated against the running matrix for active jobs using the file in the share mode. If none are found, this message is displayed.

System action: Processing continues.

User response: None.

4682 PRIOR LOCK/XLOCK HAS OCCURRED ON FILE '___'

Explanation: A lock or XLOCK has been placed on a file through an earlier job ABENDING (through SAMEND), or by a user request (SAMUPD), making it unnecessary to lock it because of the current job ABENDING.

Appendix F--Continued

System action: Processing continues.

User response: None.

4863 LOCK ON FILE '___' WILL BE INITIATED/REMOVED

Explanation:

1. When a job ABENDs, the SAMEND program will place a lock on all files used by the program that were in update, share, or exclusive-use mode.

2. When restarting an ABENDED job, the SAMEND program will remove the lock on files used by the program that were in update, share, or exclusive-use mode. A scan is initiated against the running matrix to set the current file status and number of users.

System action: Processing continues.

User response: None.

5102 OTHER KEYWORD(S) NOT PERMITTED WITH ND

Explanation: A request to update the number of devices for a file (ND) contained other keyword(s).

System action: Data statement (card) is rejected. Processing continues.

User response: Correct and re-enter.

5232 EXTRANEIOUS ',' IN COL '___'

★ Explanation: A comma was encountered on a data statement input to the MMS inquiry program with no preceding unprocessed inquiry field.

System action: Processing continues.

User response: None.

5309 PARAMETER STARTING IN COL '___' EXCEEDS MAX LENGTH

★ Explanation: A parameter on the input data statement to the MMS inquiry program exceeds the maximum length. The parameter is bypassed.

System action: Processing continues.

User response: Correct and re-enter.

Appendix F--Continued

5318 PARAMETER '___' IS INVALID

* Explanation: A parameter on the input data statement to the MMS inquiry program is invalid. The parameter is bypassed.

System action: Processing continues.

User response: Correct and re-enter.

5341 PARAMETER '___' EXCEEDS MAXIMUM

* Explanation: The parameter on the input data statement to the MMS inquiry program exceeds the maximum of 50 inquiries allowed.

System action: Processing continues.

User response: Enter a /*.

5482 SORT FAIL HAS OCCURRED

Explanation: SORT-RETURN is not equal to zero.

System action: Processing is terminated.

User response: Contact USADARCOM ALMSA (DRXAL-TA)

5568 TOTAL NUMBER OF JOBS PROCESSED TO EOJ = '___'

Explanation: Whenever the SAMEPT program is executed in batch mode, and the symbolic FUNKSHN specifies RPTALL, then the total number of jobs processed to normal completion is displayed at the end of processing.

System action: Processing continues.

User response: None.

5587 TOTAL INPUT = '___'

Explanation: When the programs SASS or SAMUPD are executing in batch mode, the total number of input records is displayed at the end of input processing.

Appendix G

MMS COMPLETION CODES

★

<u>Code</u>	<u>Definition</u>	<u>Action required</u>
100	Normal completion	None
104	No input data	Rerun with input data.
112	Simulation terminated	Correct errors reported and rerun.
116	Parameter error	Correct parameter and rerun.
120	Sort failed	Contact SMC0.

Appendix H

DREAMS READER INFORMATION

<u>Item</u>	<u>Description</u>	<u>Example</u>
001	Close the reader (usually 00C)	
002	Start DREAMSR. This will scratch data sets on the spool pack. Automatically start DREAMS that allocates a reader (usually RDR 00C).	S DREAMSR
003	To close the DREAMS reader: 1. Have end of file on. 2. Read one slash-slash card.	
004	To read in one job or the last job in a string of jobs: 1. Read one slash-asterisk card. 2. Have end of file on.	
005	After reading in 50 contiguous jobs the reader will automatically close itself. The operator must: 1. Clear the 51 job and have the job ready to be read in. 2. Start DREAMSR again as shown in item 001.	S DREAMSR
006	If non-contiguous jobs are being read in, and the total number of jobs exceeds 35, then DREAMS will close itself, reset its job count to 1, and automatically restart.	
*007	Since the DREAMS reader places all jobs on the spool pack the operator may need to start a specific job that is not under MMS control. (not on the RMATRIX).	
JM-XXXX	The operator would start RDRSAM.	S RDRSAM, XXXX=JOBNAME

Appendix I

ABEND CODES

```

*****
**                                     M M S   A B E N D S                                     **
*****
**                                     **
**                                     **
**      ABEND CODE          PROB MODULE          DESCRIPTION          **
**                                     **
**      2001          APCACCG          REGION OVERRIDE ERROR          **
**      2002          APCABBA4          NO JES2 ADDR SPACE FOUND          **
**      2003          APCABBA4          GETMAIN/SRB FAILED CSA          **
**      2003          APCABBA6          GETMAIN/SRB FAILED CSA          **
**      2004          APCACCI          REGION OVERRIDE PROBLEM          **
**      2006          APCACCI          OVER 200 DATA SETS PASSED          **
**      2007          APCACC6          NO TCAM ADDR SPACE FOUND          **
**      2010          APCACC4          BAD OPEN ON INPUT FILE          **
**      2011          APCACC4          BAD OPEN ON OUTPUT FILE          **
**      3000          APFABBB          CPU ID NOT IN SCHEDULING          **
**                                     NETWORK          **
**      3001          APCACCC          DYNAMIC ALLOCATION FAILURE          **
**      3002          APCACCC          UNABLE TO DEALLOCATE          **
**      3100          APFAAB1          BAD OPEN ON NETWORK FILE          **
**      3200          APFAAB1          DATA MISSING ON NETWORK FILE          **
**      3300          APFAAB1          BAD OPEN ON NETWORK FILE          **
**      3400          APFAAB1          DATA MISSING ON JOB FILE          **
**      3500          APFAAB1          BAD OPEN ON JOB FILE          **
**      3550          APFAAB1          LOGIC ERROR          **
**      3600          APFAAB2          OPEN ERROR ON RUN MATRIX          **
**      3700          APFAAB2          DATA MISSING ON RUN MATRIX          **
**      3800          APFAAB2          BAD OPEN ON BMATRIX          **
**      3800          APFAAB3          BAD OPEN ON DMATRIX          **
**      3810          APFAAB3          BAD OPEN ON DMATRIX          **
**      3850          APFAAB2          LOGIC ERROR          **
**      4093          APCAOB          PDS DCB ERROR (CHK R15)          **
**                                     **
*****

```

PAGE

(FILE NAME 7 J)

Appendix J

MVS LIBRARIES AND DATA SETS

ASM H V 02 14.12 07/23/65

```

21 *****
22 *****
23 *****
24 *****
25 *****
26 *****
27 *****
28 *****
29 *****
30 *****
31 *****
32 *****
33 *****
34 *****
35 *****
36 *****
37 *****
38 *****
39 *****
40 *****
41 *****
42 *****
43 *****
44 *****
45 *****
46 *****
47 *****
48 *****
49 *****
50 *****
51 *****
52 *****
53 *****
54 *****
55 *****
56 *****
57 *****
58 *****
59 *****
60 *****
61 *****
62 *****
63 *****
64 *****
65 *****
66 *****
67 *****
68 *****
69 *****
70 *****
71 *****
72 *****
73 *****
74 *****
75 *****
76 *****

```

MULTI MACHINE SCHEDULE
AUTHORIZATIONS
SECURITY
ALSA AUTALIS

S A P E N D M (A P C A 9 B)

APCABL MAIN CONTROL
* APFA03A MVS INTERFACE
* APFA081 MESSAGE & REPORT GENERATOR
* LINKED WITH SAMPENDM

DYNAMICALLY LOADED MODULES
APFA084 SAM INTERFACE
APCAB04 MVS RESOURCES
APCAB05 MVS TERMINAL SECURITY
APCAB06 Q/S TERMINAL SECURITY
APCAB0VS TERMINAL SRB

APFA081A REPCRTS
APFA082 SASS INTERFACE
APFA081 REAC AND WRITE MASTER MATRIX
APCA082 REAC AND WRITE RUN MATRIX

M A S F P T S

A P C A C B L (D R E A M S)

DYNAMICALLY LOADED MODULES
APCACC2

A P C A U B (S A M A U T O)

A P C A N B (S A M O I S P)

DYNAMICALLY LOADED MODULES
APCANBA READS VTCC IN MVT SYSTEM
APCANBVS READS VTCC IN MVS SYSTEM

A P C A R B (S A M S T A T U S)

[illegible]

ASV H V 02 19.12 07/23/85

Appendix J--Continued

LINE	TEXT	ADDRESS	HEX
1	*****	00000000	00000000
2	*****	00000001	00000001
3	*****	00000002	00000002
4	*****	00000003	00000003
5	*****	00000004	00000004
6	*****	00000005	00000005
7	*****	00000006	00000006
8	*****	00000007	00000007
9	*****	00000008	00000008
10	*****	00000009	00000009
11	*****	0000000A	0000000A
12	*****	0000000B	0000000B
13	*****	0000000C	0000000C
14	*****	0000000D	0000000D
15	*****	0000000E	0000000E
16	*****	0000000F	0000000F
17	*****	00000010	00000010
18	*****	00000011	00000011
19	*****	00000012	00000012
20	*****	00000013	00000013
21	*****	00000014	00000014
22	*****	00000015	00000015
23	*****	00000016	00000016
24	*****	00000017	00000017
25	*****	00000018	00000018
26	*****	00000019	00000019
27	*****	0000001A	0000001A
28	*****	0000001B	0000001B
29	*****	0000001C	0000001C
30	*****	0000001D	0000001D
31	*****	0000001E	0000001E
32	*****	0000001F	0000001F
33	*****	00000020	00000020
34	*****	00000021	00000021
35	*****	00000022	00000022
36	*****	00000023	00000023
37	*****	00000024	00000024
38	*****	00000025	00000025
39	*****	00000026	00000026
40	*****	00000027	00000027
41	*****	00000028	00000028
42	*****	00000029	00000029
43	*****	0000002A	0000002A
44	*****	0000002B	0000002B
45	*****	0000002C	0000002C
46	*****	0000002D	0000002D
47	*****	0000002E	0000002E
48	*****	0000002F	0000002F
49	*****	00000030	00000030
50	*****	00000031	00000031
51	*****	00000032	00000032
52	*****	00000033	00000033
53	*****	00000034	00000034
54	*****	00000035	00000035
55	*****	00000036	00000036
56	*****	00000037	00000037
57	*****	00000038	00000038
58	*****	00000039	00000039
59	*****	0000003A	0000003A
60	*****	0000003B	0000003B
61	*****	0000003C	0000003C
62	*****	0000003D	0000003D
63	*****	0000003E	0000003E
64	*****	0000003F	0000003F
65	*****	00000040	00000040
66	*****	00000041	00000041
67	*****	00000042	00000042
68	*****	00000043	00000043
69	*****	00000044	00000044
70	*****	00000045	00000045
71	*****	00000046	00000046
72	*****	00000047	00000047
73	*****	00000048	00000048
74	*****	00000049	00000049
75	*****	0000004A	0000004A
76	*****	0000004B	0000004B
77	*****	0000004C	0000004C
78	*****	0000004D	0000004D
79	*****	0000004E	0000004E
80	*****	0000004F	0000004F
81	*****	00000050	00000050

1155

Appendix J--Continued

55-621237-2385

[illegible]

[illegible]

Appendix I--Continued

151	*****	JULI MACHINE SCHEDULE	*****	01570000
152	*****	CLIST	*****	01580000
153	*****	*****	*****	01590000
154	*****	ALMS + CLIST	*****	01600000
155	*****	*****	*****	01610000
156	*****	NAME	*****	01620000
157	*****	DREAMST	*****	01630000
158	*****	*****	*****	01640000
159	*****	SAMLE	*****	01650000
160	*****	SAMLEND	*****	01660000
161	*****	SAMLEND	*****	01670000
162	*****	SAMLEND	*****	01680000
163	*****	SAMLEND	*****	01690000
164	*****	SAMLEND	*****	01700000
165	*****	SAMLEND	*****	01710000
166	*****	SAMLEND	*****	01720000
167	*****	SAMLEND	*****	01730000
168	*****	SAMLEND	*****	01740000
169	*****	SAMLEND	*****	01750000
170	*****	SAMLEND	*****	01760000
171	*****	SAMLEND	*****	01770000
172	*****	SAMLEND	*****	01780000
173	*****	SAMLEND	*****	01790000
174	*****	SAMLEND	*****	01800000
175	*****	SAMLEND	*****	01810000
176	*****	SAMLEND	*****	01820000
177	*****	SAMLEND	*****	01830000
178	*****	SAMLEND	*****	01840000
179	*****	SAMLEND	*****	01850000
180	*****	SAMLEND	*****	01860000
181	*****	SAMLEND	*****	01870000
182	*****	SAMLEND	*****	01880000
183	*****	SAMLEND	*****	01890000
184	*****	SAMLEND	*****	01900000
185	*****	SAMLEND	*****	01910000
186	*****	SAMLEND	*****	01920000
187	*****	SAMLEND	*****	01930000
188	*****	SAMLEND	*****	01940000
189	*****	SAMLEND	*****	01950000
190	*****	SAMLEND	*****	01960000
191	*****	SAMLEND	*****	01970000
192	*****	SAMLEND	*****	01980000
193	*****	SAMLEND	*****	01990000
194	*****	SAMLEND	*****	02000000
195	*****	SAMLEND	*****	02010000
196	*****	SAMLEND	*****	02020000
197	*****	SAMLEND	*****	02030000
198	*****	SAMLEND	*****	02040000
199	*****	SAMLEND	*****	02050000
200	*****	SAMLEND	*****	02060000
201	*****	SAMLEND	*****	02070000
202	*****	SAMLEND	*****	02080000
203	*****	SAMLEND	*****	02090000
204	*****	SAMLEND	*****	02100000
205	*****	SAMLEND	*****	02110000
206	*****	SAMLEND	*****	02120000
207	*****	SAMLEND	*****	02130000
208	*****	SAMLEND	*****	02140000
209	*****	SAMLEND	*****	02150000
210	*****	SAMLEND	*****	02160000
211	*****	SAMLEND	*****	02170000
212	*****	SAMLEND	*****	02180000
213	*****	SAMLEND	*****	02190000
214	*****	SAMLEND	*****	02200000
215	*****	SAMLEND	*****	02210000
216	*****	SAMLEND	*****	02220000
217	*****	SAMLEND	*****	02230000
218	*****	SAMLEND	*****	02240000
219	*****	SAMLEND	*****	02250000
220	*****	SAMLEND	*****	02260000
221	*****	SAMLEND	*****	02270000
222	*****	SAMLEND	*****	02280000
223	*****	SAMLEND	*****	02290000
224	*****	SAMLEND	*****	02300000
225	*****	SAMLEND	*****	02310000
226	*****	SAMLEND	*****	02320000
227	*****	SAMLEND	*****	02330000
228	*****	SAMLEND	*****	02340000
229	*****	SAMLEND	*****	02350000
230	*****	SAMLEND	*****	02360000
231	*****	SAMLEND	*****	02370000
232	*****	SAMLEND	*****	02380000
233	*****	SAMLEND	*****	02390000
234	*****	SAMLEND	*****	02400000
235	*****	SAMLEND	*****	02410000
236	*****	SAMLEND	*****	02420000
237	*****	SAMLEND	*****	02430000
238	*****	SAMLEND	*****	02440000
239	*****	SAMLEND	*****	02450000
240	*****	SAMLEND	*****	02460000
241	*****	SAMLEND	*****	02470000
24				

1-9

PAGE 5

Appendix J--Continued

ASP M V 02 14.12 07/23/89

```

*****
MULTI MACHINE SCHEDULE
*****
101 **
102 **
103 **
104 **
105 **
106 **
107 **
108 **
*****
NAME
*****
APC003 (HSP READER) APC003
*****
01C60000
01C70000
01C80000
01C90000
01D00000
01D10000
01D20000
01D30000
01D40000

```


PAGE 3

M 4 5 C O U N T A T I O N (R E L E A S E 7 3)

Appendix J--Continued

ASP F V Q2 07.22 07/24/85

```

53 *****
59 MULTI MACHINE SCHEDULER
60 *****
61 DATA SETS
62 *****
63 *****
64 *****
65 *****
66 *****
67 *****
68 *****
69 *****
70 *****
71 *****
72 *****
73 *****
74 *****
75 *****
76 *****
77 *****
78 *****
79 *****
80 *****
81 *****
82 *****
83 *****
84 *****
85 *****
86 *****
87 *****
88 *****
89 *****
90 *****
91 *****
92 *****
93 *****
94 *****
95 *****
96 *****
97 *****
98 *****
99 *****
100 *****
101 *****
102 *****

```

ORGANIZATION
 CONTENTS
 DATA SET NAME
 DDNAME

PARTITION DATA SET
 ALL JOBS THAT REQUIRE UPDATES TO JCL
 C.SCRAM.EXECJCL/ALMSA.MMS.EXECJCL

VOLUME
 ALMSA
 ARRCOM
 CECOM
 MICON
 TACOM
 TSARCOM
 EMRA

MISC02
 ALPH01
 2MTR1
 WTPK03
 TACOM0
 DREAMS (SYS1) USER02 (SYS3)
 VSINK3

PERMANENT JCL

ORGANIZATION
 CONTENTS
 DATA SET NAME
 DDNAME

PARTITION DATA SET
 ALL JOBS THAT DO NOT REQUIRE UPDATES
 TO JCL.
 C.SAMJCL/ALMSA.MMS.SAMJCL
 SAMJCL

VOLUME
 ALMSA
 ARRCOM
 CECOM
 MICON
 TACOM
 TSARCOM
 EMRA

MISC02
 ALPH01
 2MTR1
 WTPK03
 TACOM0
 DREAMS (SYS1) USER02 (SYS3)
 VSINK3

END

Appendix K

REF: A S F 7 0)

[illegible]

4 S MODULES (REL 1 A S E 7 J)

Appendix K--Continued

PAGE 3

ASM H V 02 14.11 07/23/85

```

32 *****
33 MULTI MACHINE SCHEDULE *****
34 PROCESS ALLOC / MODULES *****
35 *****
36 NAME DESCRIPTION LANGUAGE *****
37 *****
38 *****
39 SASS (A P F A A 3) *****
40 *****
41 MAIN CONTROL C09DL *****
42 READ AND WRITE MATRIX ASSEMBLER *****
43 READ AND WRITE MATRIX, BMATRIX ASSEMBLER *****
44 WRITE CHECK MATRIX ASSEMBLER *****
45 SASS INTERFACE *****
46 *****
47 DYNAMICALLY LOADED MODULES *****
48 APFA89A VALIDATION CONTROL *****
49 APFA89A MESSAGE & RPT GENERATOR *****
50 APFA89B REPORTS INTERFACE *****
51 APFA89C SIMULATOR *****
52 APFA89D REPORT PROCESSOR *****
53 APFA89E PRE-SIMULATION *****
54 APFA89F *****
55 *****
56 S A M E N D (A P F A 8 3) *****
57 *****
58 MAIN CONTROL C08DL *****
59 HMS INTERFACE ASSEMBLER *****
60 MESSAGE & REPORT GENERATOR C08DL *****
61 *****
62 DYNAMICLY LOADED MODULES *****
63 APFA89B SAM INTERFACE *****
64 APC88A4 MVS RESOURCES *****
65 APC88A5 MVS TERMINAL SECURITY *****
66 APC88A6 Q/S TERMINAL SECURITY *****
67 *****
68 S A M R P T (A P F A G 8) *****
69 *****
70 SAM REPORT C08DL *****
71 MESSAGE & REPORT GENERATOR C08DL *****
72 READ AND WRITE MASTER MATRIX ASSEMBLER *****
73 READ AND WRITE RUN MATRIX ASSEMBLER *****
74 *****
75 S A M U P D (A P F A F 8) *****
76 SAM UPDATE CONTROL C08DL *****
77 MESSAGE & REPORT GENERATOR C08DL *****
78 HMS INTERFACE *****
79 READ AND WRITE RUN MATRIX ASSEMBLER *****
80 *****

```

58/62/40 11.01 20 A H MSV

LINE	NAME	DESCRIPTION	LANGUAGE
82	MULTI MACHINE SCHEDULER		
83	PROCESS BLOCK / MODULES		
84			
85			
86			
87			
88			
89			
90			
91			
92			
93			
94			
95			
96			
97			
98			
99			
100			
101			
102			
103			
104			
105			
106			
107			
108			
109			
110			
111			
112			
113			
114			
115			
116			
117			
118			
119			
120			

Appendix K--Continued

58/62/23 11.01 2C A H MSV

[illegible]

Appendix K--Continued

ADDRESS	OPERATION	OPERAND	ADDRESS	OPERATION	OPERAND
156	APCASH	WRITES JOBS TO	01630000		
157		"MS JOB SPOOL/HOLD QUE	01640000		
158			01650000		
159			01660000		
160			01670000		
161			01680000		
162			01690000		
163			01700000		
164			01710000		
165			01720000		
166			01730000		
167			01740000		
168			01750000		
169			01760000		
170			01770000		
171			01780000		
172			01790000		
173			01800000		
174			01810000		
175			01820000		
176			01830000		
177			01840000		
178			01850000		
179			01860000		
180			01870000		
181			01880000		
182			01890000		
183			01900000		
184			01910000		
185			01920000		
186			01930000		
187			01940000		
188			01950000		
189			01960000		
190			01970000		
191			01980000		
192			01990000		
193			02000000		
194			02010000		
195			02020000		
196			02030000		
197			02040000		
198			02050000		
199			02060000		
200			02070000		
201			02080000		
202			02090000		
203			02100000		
204			02110000		
205			02120000		
206			02130000		
207			02140000		
208			02150000		
209			02160000		
210			02170000		
211			02180000		
212			02190000		
213			02200000		
214			02210000		
215			02220000		
216			02230000		
217			02240000		
218			02250000		
219			02260000		
220			02270000		
221			02280000		
222			02290000		
223			02300000		
224			02310000		
225			02320000		
226			02330000		
227			02340000		
228			02350000		
229			02360000		
230			02370000		
231			02380000		
232			02390000		
233			02400000		
234			02410000		
235			02420000		
236			02430000		
237			02440000		
238			02450000		
239			02460000		
240			02470000		
241			02480000		
242			02490000		
243			02500000		
244			02510000		
245			02520000		
246			02530000		
247			02540000		
248			02550000		
249			02560000		
250			02570000		
251					

NO.	NAME	DESCRIPTION	CLASS	LANGUAGE	NO.
119	APFAA8	MAIN CONTROL	ASSEMBLER	COBOL	01950000
120	APFAA9	READ AND WRITE MATRIX	ASSEMBLER	ASSEMBLER	01960000
121	APFAAB1	READ AND WRITE MATRIX	ASSEMBLER	ASSEMBLER	01970000
122	APFAAB2	WRITE CHK MATRIX	ASSEMBLER	ASSEMBLER	01980000
123	APFAAB3	VALIDATION CONTROL	ASSEMBLER	ASSEMBLER	01990000
124	APFAA9A	SASS INTERFACE	ASSEMBLER	ASSEMBLER	02000000
125	APFAAB4	MESSAGE & RPT GENERATOR	ASSEMBLER	ASSEMBLER	02010000
126	APFAAB5	REPORTS INTERFACE	ASSEMBLER	ASSEMBLER	02020000
127	APFAAB6	SIMULATOR	ASSEMBLER	ASSEMBLER	02030000
128	APFAAB7	REPORT PROCESSOR	ASSEMBLER	ASSEMBLER	02040000
129	APFAAB8	PRE-SIMULATION	ASSEMBLER	ASSEMBLER	02050000
130	APFAAB9	PRE-FLY	ASSEMBLER	ASSEMBLER	02060000
131	APFAABA		ASSEMBLER	ASSEMBLER	02070000
132	APFAABB		ASSEMBLER	ASSEMBLER	02080000
133	APFAABC		ASSEMBLER	ASSEMBLER	02090000
134	APFAABD		ASSEMBLER	ASSEMBLER	02100000
135	APFAABE		ASSEMBLER	ASSEMBLER	02110000
136	APFAABF		ASSEMBLER	ASSEMBLER	02120000
137	APFAABG		ASSEMBLER	ASSEMBLER	02130000
138	APFAABH		ASSEMBLER	ASSEMBLER	02140000
139	APFAABI		ASSEMBLER	ASSEMBLER	02150000
140	APFAABJ		ASSEMBLER	ASSEMBLER	02160000
141	APFAABK		ASSEMBLER	ASSEMBLER	02170000
142	APFAABL		ASSEMBLER	ASSEMBLER	02180000
143	APFAABM		ASSEMBLER	ASSEMBLER	02190000
144	APFAABN		ASSEMBLER	ASSEMBLER	02200000
145	APFAABO		ASSEMBLER	ASSEMBLER	02210000
146	APFAABP		ASSEMBLER	ASSEMBLER	02220000
147	APFAABQ		ASSEMBLER	ASSEMBLER	02230000
148	APFAABR		ASSEMBLER	ASSEMBLER	02240000
149	APFAABS		ASSEMBLER	ASSEMBLER	02250000
150	APFAABT		ASSEMBLER	ASSEMBLER	02260000
151	APFAABU		ASSEMBLER	ASSEMBLER	02270000
152	APFAABV		ASSEMBLER	ASSEMBLER	02280000
153	APFAABW		ASSEMBLER	ASSEMBLER	02290000
154	APFAABX		ASSEMBLER	ASSEMBLER	02300000
155	APFAABY		ASSEMBLER	ASSEMBLER	02310000
156	APFAABZ		ASSEMBLER	ASSEMBLER	02320000
157	APFAABA		ASSEMBLER	ASSEMBLER	02330000
158	APFAABB		ASSEMBLER	ASSEMBLER	02340000
159	APFAABC		ASSEMBLER	ASSEMBLER	02350000
160	APFAABD		ASSEMBLER	ASSEMBLER	02360000
161	APFAABE		ASSEMBLER	ASSEMBLER	02370000
162	APFAABF		ASSEMBLER	ASSEMBLER	02380000
163	APFAABG		ASSEMBLER	ASSEMBLER	02390000
164	APFAABH		ASSEMBLER	ASSEMBLER	02400000
165	APFAABI		ASSEMBLER	ASSEMBLER	02410000
166	APFAABJ		ASSEMBLER	ASSEMBLER	02420000
167	APFAABK		ASSEMBLER	ASSEMBLER	02430000
168	APFAABL		ASSEMBLER	ASSEMBLER	02440000
169	APFAABM		ASSEMBLER	ASSEMBLER	02450000
170	APFAABN		ASSEMBLER	ASSEMBLER	02460000
171	APFAABO		ASSEMBLER	ASSEMBLER	02470000
172	APFAABP		ASSEMBLER	ASSEMBLER	02480000
173	APFAABQ		ASSEMBLER	ASSEMBLER	02490000
174	APFAABR		ASSEMBLER	ASSEMBLER	02500000
175	APFAABS		ASSEMBLER	ASSEMBLER	02510000
176	APFAABT		ASSEMBLER	ASSEMBLER	02520000
177	APFAABU		ASSEMBLER	ASSEMBLER	02530000
178	APFAABV		ASSEMBLER	ASSEMBLER	02540000
179	APFAABW		ASSEMBLER	ASSEMBLER	02550000
180	APFAABX		ASSEMBLER	ASSEMBLER	02560000
181	APFAABY		ASSEMBLER	ASSEMBLER	02570000
182	APFAABZ</				

ASM H V 22 14.11 07/23/85

215	*****	MULTI MACHINE SCHEDULER	*****	02250000
217	**		**	02260000
218	**	MODULES	**	02270000
219	*****		*****	02280000
220	**		**	02290000
221	**	NAME DESCRIPTION LANGUAGE	**	02300000
222	*		*	02310000
223	**		**	02320000
224	**	JCL SCAN	COBOL	02330000
225	**	SEARCH POS	COBOL	02340000
226	**	GET DEVICE TYPE	ASSEMBLER	02350000
227	**	MICON ACCTN MODULE	ASSEMBLER	02360000
228	**	UPDATE RUN MATRIX	COBOL	02370000
229	**	SCAN JOB CARD	ASSEMBLER	02380000
230	**	SCAN EXEC CARD	ASSEMBLER	02390000
231	**	SCAN PROC CARD	ASSEMBLER	02400000
232	**	SCAN DD CARD	ASSEMBLER	02410000
233	**		**	02420000
234	**	DREAMS CONTROL MODULE	COBOL	02430000
235	**	SASS INQUIRY MODULE	COBOL	02440002
236	**	SAM UPDATE CONTROL	COBOL	02450002
237	**	SAM REPORT MODULE	COBOL	02460002
238	**	PROCESS INPUT DATA	COBOL	02470002
239	**	HGT REPORT MODULE	COBOL	02480000
240	**	MATRIX JOB CONVERT	ASSEMBLER	02490002
241	**	MATRIX NETWORK CONVERT	COBOL	02500002
242	**	MATRIX CONVERSION	ASSEMBLER	02510002
243	**	POST STATS TO MATRIX	COBOL	02520000
244	**	SCRATCH DSNAMES ON SPOOL	ASSEMBLER	02530000
245	**	LIST DSNAMES ON SPOOL	ASSEMBLER	02540000
246	**	READ VTOC ON SPOOL	ASSEMBLER	02550000
247	**	WRITE JCL TO POS	ASSEMBLER	02560000
248	**	DISPLAY SPOOL, HQ, EXECJCL	COBOL	02570000
249	**	WRITE EXECJCL TO SPOOL	COBOL	02580000
250	**	SCRATCH DATASETS/ST DREAMS	ASSEMBLER	02590000
251	**	SCAN RUN MATRIX	ASSEMBLER	02600000
252	**		**	02610000
253	*****		*****	02620001
254	*****	END	*****	

DATA BASE DOCUMENTATION

Appendix L--Continued

LOC	LOC	DATA FIELD	LENGTH	LOC	LOC	DATA FIELD	LENGTH
38	1-9	JOB NAME	8	38	00380004		
39	9-10	APPLICATION VOL NUMBER	2	39	00390007		
40	11-34	JOB DEPENDENCIES (12 X 2)	24	40	00400006		
41	35-36	CYCLE (REGION)	2	41	00410006		
42	37-60	FILE DEPENDENCIES (12 X 2) (SURE)	24	42	00420006		
43	61-66	DEVICE DEPENDENCIES	6	43	00430006		
44	67-68	RECENT RUN DATE	2	44	00440006		
45	69-70	RECENT RUN TIME	2	45	00450006		
46	71-72	Avg RUN TIME	2	46	00460006		
47	73-74	NUMBER OF TIMES RUN	2	47	00470006		
48	75	PRIORITY	1	48	00480006		
49	76	SIM FLAG	1	49	00490006		
50	77-84	MACHINE DEPENDENCIES	8	50	00500006		
51	85	PRIVACY	1	51	00510006		
52	86	CRITICAL PATH	1	52	00520006		
53	87	RESERVED	1	53	00530006		
54	88	SIM HOLD	1	54	00540006		
55	89-90	JOB-ES-PRIME	12	55	00550006		
56				56	00560006		
57				57	00570006		
58				58	00580006		
59				59	00590006		
60				60	00600006		
61				61	00610006		
62				62	00620006		
63				63	00630007		
64				64	00640006		
65				65	00650007		
66				66	00660013		

Appendix L--Continued

CS, Vol. 15, CCS901 18-320

ASM N V 02 11.10 07/26/75

114 NETWORK FILE 01140013
115 01150013
116 01160013

118 INSTALLATION SEGMENTS 01180006
119 01190022
120 01200006
121 01210006
122 DATA FIELD 01220006
123 INSTALLATION CODE 01230006
124 1-2 01240022
125 3-16 01250006
126 17-24 01260022
127 01270006
128 TOTAL LENGTH 01280006
129 24 01290006
130 01300006

132 MACHINE NETWORK 01320006
133 01330013
134 SEGMENT 01340006
135 01350022
136 01360006
137 01370006
138 01380006
139 01390006
140 LOCATION 01400006
141 1-2 01410013
142 3-4 01420013
143 5-6 01430013
144 7-8 01440013
145 9 01450013
146 10 01460013
147 TOTAL LENGTH 01470013
148 80 01480006
149 01490006

151 EIGHT MACHINES 01510006
152 NAME 01520013
153 CPU ID 01530013
154 AVAILABLE START TIME 01540013
155 AVAILABLE END TIME 01550013
156 RESERVED 01560013
157 TAPE USE 01570013
158 DATE OF LAST UPDATE 01580013
159 TIME OF LAST UPDATE 01590013
160 CPU ID UPDATE PROCESSED ON 01600013
161 DATE OF LAST FILE RESEV 01610013
162 TIME OF LAST FILE RESEV 01620013
163 CPU ID FILE RESEV PROCESSED ON 01630013
164 FILE STATUS SWITCH 01640013
165 (DISP 12,459 IN SECOND BLOCK) 01650013
166 UPDATED BY SCHEDULING FUNCTION 01660013

DATA BASE DOCUMENTATION

PAGE 6

Appendix L--Continued

ASM N V 02 11.10 07/20/05

167 00	(POST / UPDATE)	00	01670017
168 00		00	01680017
169 00	TOTAL SEGMENT LENGTH	14	01690015
170 00		00	01700006
172 00		00	01720013
173 00	RESERVED SEGMENT	00	01730013
174 00	TOTAL SEGMENT LENGTH	540	01740017
175 00		00	01750013

1.-4,

Appendix L--Continued

```

191 .....
192 .....
193 .....
194 .....
195 .....
196 .....
197 .....
198 .....
199 .....
200 .....
201 .....
202 .....
203 .....
204 .....
205 .....
206 .....
207 .....
208 .....
209 .....
210 .....
211 .....
212 .....
213 .....
214 .....
215 .....

```

ASM N V 02 11.10 07/26/95
 R U N M A T R I X
 RUNNING MATRIX FILE STATISTICS -
 RECORD SIZE - 100
 BLOCK SIZE (100 X 130)
 RECORDS PER BLOCK - 130
 BLOCKS WITHIN FILE - 5
 RECORDS IN FILE (130 X 5)
 TOTAL SIZE OF RUNNING MATRIX - 650
 TOTAL TRACKS - 5
 RUNNING MATRIX ELEMENT SIZE
 JOB SEGMENT (630 X 100) - 63000
 FILE SEGMENT (255 X 7) - 1785
 DESVD
 PHETDORR SEGMENT (8 X 11)
 PHATRIH WISC
 FILE CONTROL SEGMENT
 DESVD
 65000

Appendix L--Continued

[illegible]

DATA BASE DOCUMENTATION

Appendix 1--Continued

ASN N V 02 11-10 07/26/00

```

250 ..... FILE SEGMENT ..... 02500000
251 ..... 00 ..... 02510000
252 ..... 00 ..... 02520000
253 ..... 00 ..... 02530000
254 ..... 00 ..... 02540000
255 ..... 00 ..... 02550000
256 ..... 00 ..... 02560000
257 ..... 00 ..... 02570000
258 ..... 00 ..... 02580000
259 ..... 00 ..... 02590000
260 ..... 00 ..... 02600000
261 ..... 00 ..... 02610000
262 ..... 00 ..... 02620000
263 ..... 00 ..... 02630000
264 ..... 00 ..... 02640000
265 ..... 00 ..... 02650000

```

DATA FIELD	LOCATION	FILE STATUS (SUME)	FILE TYPE	NUMBER OF SCHEDULED USERS	NUMBER OF CURRENT USERS	NUMBER OF CURRENT USERS REFERENCE	NUMBER OF ABENDED JOBS USING FILE	MACHINE ID	ONE SEGMENT LENGTH	TOTAL SEGMENT LENGTH
1	1								1	1795
2	2								1	
3	3								1	
4	4								1	
5	5								1	
6	6								1	
7	7								1	

```

267 ..... RESVD ..... 02670000
268 ..... 00 ..... 02680000
269 ..... 00 ..... 02690000

```

```

271 ..... NETWORK SEGMENT ..... 02710000
272 ..... 00 ..... 02720000
273 ..... 00 ..... 02730000
274 ..... 00 ..... 02740000
275 ..... 00 ..... 02750000
276 ..... 00 ..... 02760000
277 ..... 00 ..... 02770000
278 ..... 00 ..... 02780000
279 ..... 00 ..... 02790000
280 ..... 00 ..... 02800000
281 ..... 00 ..... 02810000
282 ..... 00 ..... 02820000
283 ..... 00 ..... 02830000
284 ..... 00 ..... 02840000
285 ..... 00 ..... 02850000

```

DATA FIELD	LOCATION	CPU ID	CPU NONAVAIL START TIME	CPU NONAVAIL END TIME	RESERVED	TAPED USAGE EQUAL TAPE TIME	TAPED USAGE EQUAL TAPE TIME	ELEMENT MACHINES	SEGMENT LENGTH
1-2	1-2								2
3-4	3-4								2
5-6	5-6								2
7-8	7-8								1
9	9								1
10	10								1

```

289 ..... HISCATR DATA ..... 02890000
290 ..... 00 ..... 02900000
291 ..... 00 ..... 02910000
292 ..... 00 ..... 02920000
293 ..... 00 ..... 02930000
294 ..... 00 ..... 02940000
295 ..... 00 ..... 02950000
296 ..... 00 ..... 02960000
297 ..... 00 ..... 02970000
298 ..... 00 ..... 02980000
299 ..... 00 ..... 02990000

```

DATA FIELD	LOCATION	MESSAGE	DATE LAST RUN OR FLV	TIME LAST RUN OR FLV
1-9	1-9			
9-10	9-10			
11-12	11-12			

PAGE 11

Appendix L--Continued

DATA BASE DOCUMENTATION

ASN	N	Q2	11-10	07/16/95
299	00	13	-2C	02990022
300	00	21	-34	03000000
301	00	21	-34	03010000
302	00	35	-1P	03020000
303	00	35	-1P	03030000
304	00	19		03040000
305	00	5C		03050011
306	00			03060000
307	00			03070011
308	00			03080000
309	00			03090001
310	00			03100011
311	00			03110001
312	00			03120001
313	00			03130000
314	00			03140000
315	00			03150013
316	00			03160013
317	00			03170013
318	00			03180013
319	00			03190013
320	00			03200013
321	00			03210013
322	00	1-2		03220019
323	00	3-4		03230019
324	00	5-6		03240019
325	00	7-8		03250019
326	00	9-10		03260019
327	00	11-12		03270013
328	00	13		03280016
329	00	14		03290019
330	00			03300016
331	00			03310013
332	00			03320016
333	00			03330016
334	00			03340013

NAME	LENGTH
DATE OF LAST UPDATE	2
TIME OF LAST UPDATE	2
CPU ID UPDATE PROCESSED IN	2
DATE OF LAST FILE RESET	2
TIME OF LAST FILE RESET	2
CPU ID FILE RESET PROCESSED ON	2
FILE CONTROL STATUS SWITCH	1
(DISP 12-029 IN LAST DATA BLOCK)	1
UPDATED BY SCHEDULING FUNCTION	1
(SAMEID / SAMEID)	1
TOTAL SEGMENT LENGTH	14

335	00			03350000
336	00			03360000
337	00			03370019
338	00			03380000

MSM 4 V 22 11-10 07/16/85

[illegible]

INDEX

ABEND, definition of 1-1
Application, definition of 1-1
*Application matrix segment 2-3
Back-off job dependencies 2-2
Coding examples
 see SASS coding examples
Compare utility 2-19
Completion code modification 3-14
Concept
 DREAMS 4-1
 SAM 3-1
 SASS 2-1
 general 2-4
Control cards for
 master data base update 2-5
 on-the-fly update 2-16
 on-the-fly update test 2-15
 schedule initialization 2-11
 simulation 2-10
 standard reports 2-13
CORE segment 2-4
Cross-reference by usage 2-18
Data card, general 2-4
Data cards for
 master data base update 2-5
 on-the-fly update 2-16
 running matrix update 3-8
Dependency relationships
 back-off of 2-2
 no-back-off 2-2
Disk checking
 SAMEND parameter for 3-15
Distribution instructions 2-20
DREAMS execution by function
 DREAMSB 4-2
 DREAMSP 4-2
 DREAMSR 4-3
 DREAMST 4-3
Dummy UCB segment 2-4
File lock
 initiating via SAMUPD 3-8
 removal via job restart 3-7
File matrix segment 2-4
File modes, definitions of
 exclusive-use 1-1
 reference 1-1
 share 1-1
 update 1-2

HASP
 definition of 1-1
Infinite dependency set-up 2-19
Initiator class 3-13
Installation segment 2-4
JOB, definition 1-1
JOB matrix segment 2-1
JOB restart 3-6
Manually releasing JOB 3-13
Master data base
 definition of 1-11
 Control card segments, descrp 2-1
Master data base reports
 cross-reference 2-14
 inquiry 2-15
 standard 2-11
Master data base update 2-5
Master matrix, definition of 1-1
No back-off dependencies 2-2
On-the-fly update
 definition of 1-1
 function 2-17
On-the-fly update test 2-15
Operational considerations
 required 3-13
 special conditions 3-13
 parameter values for SAMEND 3-15
Posting job completion 3-8
Posting statistics 3-7
Predecessor job, definition of 1-1
Quiesce
 definition of 1-1
 required operational
 considerations 3-7
Run matrix status 3-6
Running matrix
 definition of 1-1
 elements within 3-3
 save and recovery 3-14
Running matrix update 3-8
SAM execution options 3-6
SAM functions
 job restart 3-7
 posting job completion 3-8
 posting statistics 3-7
 run matrix statistics 3-6

Index--Continued

- running matrix update 3-8
- simulation of current workload 3-7
- system environment scan 3-6
- system scan 3-8
- system restart 3-7
- system termination 3-7
- SAM execution by function
- SAMABEND 3-8
- SAMCOPY 3-13
- SAMDEL 3-13
- SAMEDIT 4-4
- SAMEND 3-8
- SAMENV 3-7
- SANCANC 3-12
- SAMDISP 3-12
- SAMDOWN 3-12
- SAMLIST 3-13
- SAMNAMES 3-13
- SAMPOST 3-8
- SAMREST 3-7
- SAMRPTA 3-5
- SAMRPTI 3-6
- SAMRPTIH 3-6
- *
- SAMRPTP 3-6
- SAMRPTR 3-6
- SAMRUN 3-7
- SAMSCAN 3-9
- *SAMSIMP 3-8
- *SAMSPool 3-13
- SAMST 3-13
- SAMSTART 3-13
- SAMTERM 3-7
- SAMUPD 3-9
- SAMEND parameter values 3-14
- SASS coding examples
 - cross-reference reports 2-21
 - initiating a schedule 2-21
 - on-the-fly updating 2-20
- * simulation 2-20
 - updating segments 2-20
- SASS execution by function
- SASSCHK 2-15
- SASSFLY 2-16
- SASSFMT 2-5
- SASSRPT 2-13
- SASSRUN 2-12
- SASSSIM 2-11
- SASSUPD 2-5
- SCAMINQ 2-14
- SCANINQ 2-14
- SCCOMPX 2-18
- SCJMINQ 2-14
- SCJNINQ 2-14
- SCXREF 2-14
- SASS functions
 - list of 2-1
 - master data base format 2-5
 - master data base reports
 - cross-reference 2-13
 - inquiry 2-13
 - standard 2-11
 - master data base update 2-5
 - on-the-fly update 2-16
 - on-the-fly update test
 - restrictions 2-15
 - sequence of processing 2-15
 - schedule initialization 2-13
 - simulation 2-10
 - schedule initialization 2-11
 - simulation 2-10
- SASS input data set
 - control card, general 2-4
 - data card, general 2-4
- SASS reports
 - cross-reference 2-13
 - inquiry 2-15
 - standard 2-13
- SASS special options
 - distribution instructions 2-20
 - infinite dependency set-up 2-19
- SASS special utility
 - compare 2-19
- SHARDISK, definition of 1-2
- Simulation
 - function 2-10
 - reports 2-10
- Simulation of current workload 3-7
- SVC used within MMS 1-2
- System resource definition at
 - execution time 1-2
 - simulation time 1-2
- System scan 3-8
- System restart 3-7
- System termination 3-7
- Trace of actions 1-2

INDEX (continued)

TSO

command procedures 1-2
definition of 1-1
log-on procedure 1-2

THE PROPOSER AGENCY OF THIS PUBLICATION IS THE AUTOMATED LOGISTICS MANAGEMENT SYSTEMS ACTIVITY. USERS ARE INVITED TO SEND COMMENTS ON DA FORM 8000 (RECOMMENDED CHANGES TO PUBLICATIONS) TO THE COMMANDER, ALMBA, ATTN: DOTAL-AAS, PO BOX 1878, ST LOUIS, MISSOURI 63108.

(DOTAL-TA)

FOR THE COMMANDER:

OFFICIAL:


O. D. GOODWIN
Chief, Administrative Division

JAMES GARFIELD-JEFFERSON
MAJ, Inf
Acting Adjutant

DISTRIBUTION:
S

BEST
AVAILABLE COPY